

OGG THEORA

Published : 2011-03-11

License : None

JOHDANTO

1. JOHDANTO

2. Tästä käyttöoppaasta

3. Mitä on digitaalinen video?

4. MITÄ THEORA ON?

JOHDANTO

Internetin nopean kasvun tärkeimpiä taustavaikuttajia on ollut puolueeton suunnittelumalli. Avoimet standardit, vapaat ohjelmistot ja läpinäkyvät teknologiat ovat vahvistaneet verkon kehitystä. Verkon avoimen ympäristön seurauksena sisällönkehittäjät voivat helposti luoda verkkosivuja, ohjelmistojen kehittäjät voivat rakentaa olemassaolevien projektien päälle ja laajentaa toimintamahdollisuuksia verkossa, ja käyttäjät voivat selata verkkoa, lähettää sähköpostia ja käyttää monenlaisia sovelluksia.

Verkkovideon tuotannon ja esittämisen perustekniikat eivät kuitenkaan ole vapaita tai läpinäkyviä. Jos esimerkiksi ohjelmistokehittäjät tekevät uuden videosoittimen, he joutuvat maksamaan tekijänoikeusmaksuja yrityksille, jotka pitävät hallussaan tiettyjä patentteja, jos tahtovat jakaa videota laillisesti. Nämä patentit (kuten useimmat ohjelmistopatentit) lukitsevat äärimmäisen perustavanlaatuisia ideoita. Esimerkiksi: yksi yhtiö on patentoinut idean tallentaa kuvan osia vasemmalta oikealle, ylhäältä alas! Yhdessä tuhannet videotekniikoita koskevat patentit estävät uudet tavat jaella mediaa ja tehdä siitä interaktiivista.

PATENTIN KUORMITTAMA VIDEO

Niille, jotka ovat huolestuneita yksityisten etujen vallasta julkisessa elämässä, tämä on ilmeinen ongelma: kuinka pieni joukko yhtiöitä voi hallita ohjelmistojen kehittämiseen tarvittavia perusideoita? Se luo myös hyvin konkreettisia ongelmia: maailmanlaajuinen vapaiden ohjelmistojen liike ei voi liittää monien videoformaattien tukea tuotteisiinsa. Ja koska patentit tekevät omien videosoittimien levittämisestä kallista ohjelmistojen kehittäjille, kehittäjät ovat riippuvaisia muutamista Microsoftin, Applen ja Adoben tuotteista. Verkossa yli 98% videosta jaellaan Flashia käyttäen.

Ongelmia ei ole vain ohjelmistokehityksen maailmassa - jos luot tai julkaiset videon verkossa, patenttien kuormittamat teknologiat vaikuttavat sinuun suoraan. Patenttien omistajilla on valta ottaa maksuja sisällöntuottajilta ja MPEG LA, joka edustaa suosittua MPEG-videoteknologian patenttien omistajia, ottaa lisenssimaksuja televisioyhtiöiltä, DVD-jakelijoilta, ja muilta. Tällä hetkellä he eivät vaadi rahaa verkkolähetyksistä, mutta uudet tekijänoikeusmaksuehdot voivat jossain vaiheessa uhata verkkovideon riippumattomia julkaisijoita.

PAREMPI TAPA

Nyt meillä on kuitenkin verkkovideoteknologia, jota kuka tahansa on vapaa käyttämään, tutkimaan, parantamaan ja jakelemaan ilman lupaa tai maksua. Tätä teknologiaa kutsutaan Ogg Theoraksi (tai vain "Theoraksi"). Jotkin osat Theorasta on parentoitu, mutta patenttien omistajat ovat antaneet pysyvän, peruuttamattoman, tekijänoikeusmaksuvapaan patenttilisenssin kaikille. Theora välttää tarkkaan kaikkia patentteja, jotka ovat perinteisillä patenttien hallitsijoilla. Esimerkiksi kiertääkseen naurettavaa yllä mainittua kuvien tallennukseen liittyvää patenttia, Theora tallentaa videokuvatietoa alhaalta ylös eikä ylhäältä alas!

On tärkeää ymmärtää, että Theora on merkittävä osa internetin vapautta. Mozilla, Opera ja Google ymmärtävät tämän, ja ovat julistaneet tukevansa Theora-videota tulevaisuudessa tai nykyisissä selainversioissaan. Tämä merkitsee, että miljoonat käyttäjät kykenevät katsomaan Theora-videota käyttäen selaintaan, ilman ylimääräisiä ohjelmistoja. Vapaiden ohjelmistojen yhteisön työ, Mozillan, Wikipedian ja muiden tukemana, on tuonut Theoran samalle laatutasolle kuin uusimman teknologian videotekniikat.

On myös muita tärkeitä projekteja, joilla on samanlaiset päämäärät, kuten Dirac, jonka kehitystä BBC johtaa. Mutta jännittävintä Theorassa on, että se on täällä nyt, sitä tukevat suosittu työkalut, ja se on valmis laajamittaiseen käyttöön. Opettelemalla Theoraa ja käyttämällä sitä omissa projekteissasi voit tehdä verkosta jännittävemmän ja vapaamman.

2. TÄSTÄ KÄYTTÖOPPAASTA

Tämä käyttöopas kirjoitettiin viiden päivän kirjapyrähdyksen aikana Berliinin Maailmankulttuurin talossa (elokuun 10-15 2009).



Paikalla olivat:

Jan Gerber

Homes Wilson

Susanne Lang

David Kühling

Jörn Seger

Ja monet auttoivat verkossa. Kiitokset myös Leslie Hawthornille ja Googlelle pyrähdyksen tukemisesta, ja Berliinin kesäleirille. Tämä kirja on jatkuvat prosessi... Ole hyvä ja parantele sitä!

1. REKISTERÖIDY

Rekisteröidy at FLOSS Manuals:

<http://fi.flossmanuals.net/register>

2. KIRJOITA!

Valitse käyttöopas <http://fi.flossmanuals.net/bin/view/TheoraCookbook/WebHome> ja luku jota tahdot muokata.

Jos tahdot kysyä meiltä kysymyksiä käyttöoppaiden kirjoittamisesta, liity alla mainitulle irc-kanavalle ja kysy meiltä! Odotamme osallistumistasi innokkaasti!

Lisää tietoa FLOSS Manualsista löydät omasta käyttöoppaastamme:

<http://fi.flossmanuals.net/FLOSSManuals>

3. IRC

On hyvä jutella kanssamme, jotta voimme koordinoida kaikkia projektiin osallistuvia. Meillä on IRC-kanava jolle voit liittyä.

Ota IRCissä yhteys seuraaviin:
palvelin: irc.freenode.net
kanava: #flossmanuals

4. POSTILISTA

Kaikkea FLOSS Manuals -aiheista keskustelua varten liity postilistallemme:
<http://lists.flossmanuals.net/listinfo.cgi/discuss-flossmanuals.net>

3. MITÄ ON DIGITAALINEN VIDEO?

Video on sarja kuvia, jotka näkyvät liikkuvana kuvana. Ensimmäiset "videot" (elokuvat) olivat kirjaimellisesti sarja valokuvia, jotka oli valaistu yksi kerrallaan tarpeeksi nopeasti, jotta ihmisen silmä näki liikkeen. Digitaalista videota voidaan ajatella kuvien sarjana, mutta yleensä todellisuus on monimutkaisempi.

Videota voidaan varastoida sarjana kuvia, aivan kuin filminauha tekee, mutta tämä on jonkin verran tuhlaavaista. Loppujen lopuksi liikkuvan kuvan osat eivät usein muutu ruudusta toiseen, tai sitten ne muuttuvat hyvin vähän. Näissä tapauksissa on helpompaa kuvata pelkkää muutosta.

Itse asiassa, näin videoformaatit yleensä toimivat. Maailmassa, jossa on rajoittunut määrä levytilaa ja verkkoyhteyksiä tämän kaltaiset yksinkertaiset ideat voivat antaa sinun tallentaa satoja videoita tietokoneellesi (kourallisen sijasta) tai ladata videon minuuteissa (tuntien sijasta).

Edellisen tekniikan lisäksi videoformaatit voivat käyttää muita pakkausmekanismeja, kuten samanväristen alueiden kuvaamista sen sijaan, että jokainen piste kuvattaisiin yksitellen (samanlaisia temppuja voidaan käyttää kuvien kuvailuun), tai ne kuvailevat tarkasti tiettyjä kuvan osia, joissa on paljon yksityiskohtia tai liikettä tapahtuu (esimerkiksi osa, johon luultavasti tuijotat) ja keskittyy vähemmän tylsiin juttuihin reunoilla. Nämä videopakkausten ideat eivät ole uusia, mutta ne kehittyvät nopeasti.

MIKÄ THEORA ON?

Theora on videoteknologia videon luomiseen, leikkaamiseen, käsittelyyn ja toistamiseen. Tämän tyyppisestä teknologiasta puhutaan usein videoformaattina tai kodekkina (yhdistelmä sanoista koodaaja-dekoodaaja). Theora on vapaa videoformaatti, joten kuka tahansa voi vapaasti käyttää, tutkia, parantella ja jaella sitä ilman lupaa. Jotkin osat Theorasta ovat patentoituja, mutta niiden patenttien omistajat ovat antaneet pysyvän ja peruuttamattoman tekijänoikeusmaksuttoman lisenssin kaikille.

Koska Theoran jakelua ja parantelua eivät rajoita patentit, se voidaan sisällyttää vapaisiin ohjelmistoihin. GNU/Linux pohjaisten käyttöjärjestelmien jakeluversiot, kuten Ubuntu, Debian GNU/Linux tai Fedora sisältävät Theoran valmiiksi asennettuna. Ja vapaat verkkoselaimet, kuten Firefox ja Chrome tukevat Theoraa. Jos ajattelemme verkkoselainten käyttötilastoja heinäkuulta 29, noin 25% netin käyttäjistä käyttää Firefoxia ja 2,5% käyttää Chromea. Tämä merkitsee, että joka päivä valtava joukko ihmisiä käyttää Theora-ohjelmien toistamiseen kykeneviä ohjelmistoja.

HISTORIA

Theora perustuu vanhempaan teknologiaan nimeltä **VP3**, joka oli alunperin yksityisomisteinen ja patentoitu videoformaatti, jonka oli kehittänyt On2 Technologies -yhtiö. Syyskuussa 2001 On2 lahjoitti VP3:n Xiph.org Foundationille vapaan ohjelmistolisenssin alaisuudessa. On2 antoi myös peruuttamattoman, tekijänoikeusmaksuttoman lisenssiluvutuksen, jossa luopui kaikista patenttivaatimuksista, joita sillä voisi olla ohjelmasta ja sen johdannaisista. Näin kuka tahansa voi rakentaa VP3-teknologian varaan ja käyttää sitä mihin tahansa tarkoitukseen. Vuonna 2002 On2 sopi Xiph.Org Foundationin kanssa, että VP3:sta tulisi perusta uudelle Theoraksi kutsutulle videoformaatille. On2 julisti Theoran olevan VP3:n seuraaja.

Xiph.Org Foundation on voittoa tuottamaton järjestö, joka keskittyy vapaiden multimediaformaattien ja -ohjelmien tuotantoon ja levittämiseen laajemmalle yleisölle. Theoran kehityksen lisäksi se on luonut vapaan äänikodekin nimeltä Vorbis, sekä suuren joukon käyttökelpoisia työkaluja ja osia, jotka tekevät vapaista multimediaohjelmista helpokäyttöisempiä.

Muutaman vuoden beta-testauksen jälkeen Theora julkaisi marraskuussa 2008 ensimmäisen vakaan version 1.0. Millä tahansa Theoran versiolla enkoodatut versiot ovat tämän pisteen jälkeen yhteensopivia minkä tahansa tulevan soittimen kanssa. Laajaa Theoran kehittäjien yhteisöä tukevat Redhatin kaltaiset yhtiöt ja kansalaisjärjestöt, kuten Wikimedia Foundation.

VERKKO

Theoran tuki verkkoselaimissa luo erityisen tilaisuuden. Juuri nyt lähes kaikki verkkovideo vaatii Flashia, yhden yhtiön omistamaa tuotetta. Mutta nyt 25% käyttäjistä voi toistaa Theora-videoita selaimessaan asentamatta lisäohjelmistoja, joten on mahdollista haastaa Flashin valta-asema verkkovideon jakelutyökaluna. Lisäksi W3C:n (World Wide Web Consortium) uusi HTML5-standardi tuo uuden jännittävän ulottuvuuden - verkon ja videon yhdistämisen monilla tavoilla, jotka sopivat Theoralle.

PATENTIT JA TEKIJÄNOIKEUDET

Patenttien maailma on monimutkainen ja jättää Theoran kilpailijoille paljon tilaa levittää pelkoa, epävarmuutta ja epäilyksiä sen käyttökelpoisuudesta todella vapaana formaattina. Olemukseltaan Theora on vapaa. Se on vapaa käytettäväksi, muutettavaksi, uudelleenjaettavaksi, toteutettavaksi, myytäväksi tai mitä tahansa sillä tahdotkin tehdä. Mutta on myös tärkeää, että vapaiden ohjelmistojen tukijat ymmärtävät esille nousevat kysymykset.

Yksi yleinen pelko on, että Theora kohtaa huomaamattomia patentteja - patentteja, joita kukaan ei ole vielä huomannut, mutta joita Theoran tekijät eivät saaneet käyttää. On kuitenkin totta, että kaikki nykyaikaiset ohjelmat voivat törmätä salattuihin patentteihin - kaikki Microsoft Wordista Linuxin ytimeen. Kuitenkin miljoonat ihmiset ja kokonaiset teollisuudenalat käyttävät vieläkin näitä työkaluja. Toisin sanottuna salatut patentit ovat ongelma koko ohjelmistoteollisuudelle, mikä ei tarkoita, että ohjelmistoteollisuuden pitäisi lopettaa ohjelmien kehittäminen.

On myös tärkeää havaita, että pahimmakin tapauksessa tuntemattoman patentin ilmestyessä Theora voi luultavasti kiertää sen. Tällaista tapahtuu koko ajan. Mozilla Foundationin ja Wikipedian kaltaiset organisaatiot ovat tutkineet tätä kysymystä ja ovat tulleet samaan lopputulokseen.

Xiph.orgin tuottamat ohjelmat ovat myös tekijänoikeuden alaisia, mutta ne on tarjottu vapaiden ohjelmistolisenssien alla. Xiph.org tarjoaa koodia, jotta jokainen voi lisätä sen mihin tahansa ohjelmaan. Xiph.org tarjoaa myös joukkoa työkaluja, jotka toimivat Theora-tiedostojen kanssa. Tämä merkitsee, että voit tutkia, muuttaa, uudelleenjaella ja myydä kaiken, mitä teet Theoralla tai millä tahansa sen sisältävillä työkaluilla.

MUUT VAPAAT VIDEOFORMAATIT

On hyvä huomata, että on myös toinen projekti, joka luo tekijänoikeusmaksutonta, kehittyntä videopakkausformaattia. Sitä kutsutaan Diraciksi. Sen loi alunperin BBC:n tutkimusosasto. Dirac yrittää tulevaisuudessa toimia kaikissa sovelluksissa Internet-streamauksesta korkean tarkkuuden televisioon asti ja laajeta toimimaan uusien laitteiden kanssa. Ogg Theora toimii silti erittäin hyvin verkon streaming-videojakelussa, vaikka Diracista tulee luultavasti parempi vaihtoehto korkean tarkkuuden videoiden jakeluun.

PERUSKÄSITTEET

5. KOODEKIT

6. SÄILÖT

KOODEKIT

Koodekki on matemaattinen kaava, joka kutistaa video- tai audiotiedoston kokoa. Theora on videokoodekki.

Kun koodekki kutistaa videotiedoston kokoa, sen sanotaan myös **tiivistävän** tiedostoa. On kaksi tiivistystapaa, jotka ovat mielenkiintoisia tässä yhteydessä - **tappioton** ja **tappiollinen tiivistys**.

1. **Tappioton tiivistys** - Tämä on datan tiedon tiivistäminen pienempään kokoon ilman datan poistoa. Visualisoidaksesi tämän prosessin kuvittele paperipussi, jonka sisällä on esine. Kun poistat ilman pussista luomalla tyhjiön, pussissa oleva esine ei kutistu, vaikka pussin koko kutistuu.
2. **Tappiollinen tiivistys** - Kutsutaan joskus "havaittavaksi koodaukseksi", tämä merkitsee datan "heittämistä pois" tiedostojen koon pienentämiseksi. Pakkausalgoritmit ovat monimutkaisia ja yrittävät säilyttää havaintokokemuksen laadun mahdollisimman pitkälle, vaikka hylkäävät niin paljon dataa kuin on tarpeen. Tappiollinen tiivistys on hyvin hienoa taidetta. Algoritmit, jotka mahdollistavat tämän, ottavat huomioon kuinka aivot havaitsevat ääntä ja kuvaa ja poistavat tietoa äänestä ja videosta, vaikka säilyttävät alkuperäistä lähdemateriaalia muistuttavan ääni- ja kuvakokemuksen. Tehdäkseen tämän prosessi seuraa psykoakustisen ja psykovisuaalisen mallinnuksen periaatteita.

LAATU

Digitaalisen videon laatua määrittää koodatun informaation määrä (bittinopeus) ja käytetyn videopakkauksen tyyppi (koodekki). Vaikka joitain koodekkeja voidaan pitää Theoraa kehittyneimpinä, laadussa havaittava ero ei ole merkittävä.

BITTINOPEUS JA LAATU

Koska digitaalinen video esittää liikkuvan kuvan informaationa, on loogista, että mitä enemmän informaatiota videossa on, sitä laadukkaampaa liikkuva kuva on. Bittinopeus on kirjaimellisesti bittien määrä sekunnissa videota (ja/tai audiota), jota käytetään koodaukseen. Koodekin suurempi bittinopeus merkitsee suurempaa laatua. Annetulle kestolle suurempi bittinopeus merkitsee myös suurempaa tiedostoa.

On siis monta syytä, miksi matalampi bittinopeus on hyvä asia. Videon täytyy ehkä mahtua tietylle tallennusmedialle, kuten DVD:lle. Tai tahdot ehkä toimittaa videon tarpeeksi nopeasti yleisöllesi (jonka keskimääräinen nettiyhteys on nopeusrajoitteinen) katsottavaksi latauksen aikana.

Erilaiset videotyypit voivat vaatia erilaisia bittinopeuksia saavuttaakseen saman havaittavan laadun. Video, jossa on paljon leikkauksia ja jatkuvasti vaihtuvia kamerakulmia, vaatii enemmän tietoa kuvaukseensa, kuin video jossa on paljon liikkumattomia kuvia. Toimintaelokuva vaatii siis suuremman bittinopeuden kuin Ken Burns'n dokumentti.

Jotkin koodekit sallivat **muuttuvan bittinopeuden**. Tämä merkitsee, että bittinopeus voi muuttua ajan myötä vastauksena vaadittuun yksityiskohtaisuuteen. Tässä tapauksessa videokoodekki käyttäisi enemmän bittejä nauhoittaakseen 10 sekuntia nopeita leikkauksia ja liikkuvia kamerakulmia, kuin se vaatisi koodataksaan 10 sekuntia melko liikkumatonta kuvaa.

KOODEKIT JA LAATU

Koodekit pienentävät mediatiedoston tarpeellista bittinopeutta kuvailemalla median ovelilla, tehokkailla tavoilla. Videokoodekit kuvaavat muutoksia yhdestä kuvaruudusta seuraavaan, sen sijaan että kuvaisivat jokaista kuvaruutua erikseen. Audokoodekit jättävät huomiotta taajuudet, joita ihmiskorva ei kuule. Aivan kuin yksinkertaiset tekniikat voivat kutistaa suuresti tiedoston bittinopeutta ja kokoa, vielä kehittyneemmät tekniikat voivat kutistaa sitä vielä enemmän. Tämän vuoksi joitain koodekkeja pidetään toisiin verrattuna ylivermaisina.

Kun koodekit käyttävät monimutkaisia matemaattisia tekniikoita videon koodaamiseen, tarvitset voimakkaan mikrosirun avataksesi koodauksen nopeasti toistoa varten. Tämän vuoksi on usein parasta käyttää yksinkertaisempaa koodekkia. Viimeisimmän teknologian mukaisia temppuja käyttävä koodekki voi olla käyttökelpoinen vanhemmalla tietokoneella. Ja joskus on vain parempi käyttää yksinkertaisempaa ja vanhempaa videokoodekkia (kuten DVD-levyt tekevät), koska sen toistoon tarvittava laitteisto on halvempi.

THEORA JA LAATU

Viimeaikaisen Theora-yhteisön työn ansiosta Theora on saavuttanut saman laatutason kuin muut modernit koodekit, kuten h.264 (Applen, Youtuben ja muiden käyttämä patenttien kuormittama koodekki). Tämä voi olla kiistanalainen asia, ja on joitain syitä pitää h.264-koodekkia teknisesti ylivermaisena Theoraan verrattuna. Parasta on tutustua asiaan itse ja muodostaa oma mielipide.

Näillä sivuilla on vertailuja Theoran ja h.264:n välillä:

<http://people.xiph.org/~greg/video/ytcompare/comparison.html>

<http://people.xiph.org/~maikmerten/youtube/>

SÄILÖT

Säilö tai **kääre** on tiedostoformaatti, joka määrittelee miten erilaiset tietovirratt voidaan varastoida yhdessä tai lähettää yhdessä verkon yli. Se sallii audio- ja videodatan tallentamisen yhteen tiedostoon ja toiston synkronisoidusti. Se sallii myös datasta etsimisen, kertomalla toisto-ohjelmalle missä audio- ja videodata on tiettynä ajan pisteinä.

Audion ja videon lisäksi säilöt voivat tarjota metadatan sisältämästään datasta, esimerkiksi kuvaruutujen koosta, kuvataajuudesta, onko ääni monoa vai stereota, näytetaajuudesta, ja myös tietoa datan koodaamiseen käytetyistä koodekeista.

Kun toistat digitaalisen elokuvan, jossa on ääntä, soittimesi lukee säilöä, ja dekodaa videon ja audion käyttäen eri koodekkeja. Theora on yleensä varastoitu tai virtauslähetetty yhdessä Vorbis-äänien kanssa Ogg-säilössä, mutta se voidaan varastoida myös muihin säilöihin. matroska (.mkv) on toinen formaatti, jota ihmiset käyttävät Theora-videon.

ERO SÄILÖN JA KOODEKIN VÄLILLÄ.

Kolmen kirjaimen loppupääte tiedostonimen lopussa viittaa säilöön, ei koodekkiin. Tämä hämmentää usein ihmisiä. Kun tiedosto loppuu .mp4 tai .avi, päätteet viittaavat säilöihin, jotka voivat sisältää monia erilaisia audio- ja videoströimien yhdistelmiä. Monet säilöt eivät toimi tietyillä koodekeilla, ja tietyt koodekit toimivat parhaiten tietyillä säilöillä. Mutta et voi sanoa varmasti mitä koodekkeja videotiedosto vaatii katsomalla tiedostopäätettä.

TOISTO

7. KUINKA THEORAA TOISTETAAN

8. VLC

9. MIRO

THEORAN TOISTAMINEN

Toistaaksesi Ogg Theora -videoita tarvitset videosoittimen, joka tukee Ogg Theoran toistoa. Usein joidenkin videotyyppien toisto vaatii utojen ohjelmien asentamista, mikä voi olla hyvin turhauttavaa ja ajanhukkaa. Onneksi monet videosoittimet voivat toistaa Ogg Theoraa ilman tarvetta asentaa mitään muuta. Kaksi helpointa soitinta ovat huomattavia, koska ne toimivat samoin kaikissa tärkeimmissä käyttöjärjestelmissä (Linux, Mac OS/X, Windows).

VLC on vapaa videosoitin, jolla toistaa monenlaisia videotiedostoja, myös Theoraa. Voit hankkia sen verkosta: <http://www.videolan.org/vlc>

Miro on toinen Ogg Theora -videota tukeva videosoitin (<http://getmiro.com>).

Jos sinulla ei ole kumpaakaan etkä tahdo asentaa niitä, voit myös käyttää Firefox-selaimen versiota 3.5 ja myöhempiä (<http://getfirefox.com>) Theoran katseluun.

Peukalosääntönä: jos video on työpöydälläsi, käytä VLC tai Miro -videosoitinta. Jos video on verkossa, käytä Firefox 3.5 tai myöhempi.

THEORAN INTEGROINTI

Jos tahdot käyttää Theoraa soittimella, joka tuli Mac OS X:n tai Windowsin mukana, kuten Windows Media Player tai QuickTime Player, sinun tulee asentaa komponentteja tai filttäreitä, jotka parantavat Theoran toimivuutta. Tämä ei päde, jos käytät Linuxia, koska Theoraa tuetaan useimmissa Linux-jakeluissa ja sitä voi toistaa Totemilla ja muilla GStreameriin perustuvilla ohjelmilla.

Windowsissa ja Mac OS X:ssä voit lisätä täyden toiminnallisuuden ja Theora-tuen kaikkiin QuickTime-pohjaisiin sovelluksiin, kuten QuickTime-soittimeen itseensä, mutta myös iTunesiin ja iMovieen. Täytyy vain asentaa Xiph QuickTime Component (XiphQt), joka voidaan ladata Xiph.Orgin verkkosivulta: <http://xiph.org/quicktime/download.html>

Toinen hyödyllinen suodatin Windows-käyttäjälle on DirectShow Filter. Se on myös tarjolla Xiph.Org Foundationista ja lisää koodauksen ja dekodauksen tuen Ogg Vorbisiin, Speexiin, Theoraan ja FLACiin mihin tahansa DirectShow -sovellukseen, kuten Windows Media Playeriin. Voit myös ladata suodattimet Xiph.Org -verkkosivulta: <http://xiph.org/dshow/>

VLC

VLC (lyhenne sanoista *VideoLAN Client*) on erinomainen työkalu video- ja audiotiedostojen toistoon. Se on vapaa ohjelmisto, se soittaa suuren määrän formaatteja, joihin sisältyy Theora, ja se toimii monilla erilaisilla alustoilla. Jos käytät Window, Mac OS/X tai Linux (esimerkiksi Ubuntu) -käyttöjärjestelmää, VLC on sinulle mainio vaihtoehto. VLC toimii samoin jokaisessa käyttöjärjestelmässä, joten jos tiedät kuinka sitä käytetään Windowsissa, osaat käyttää sitä myös Ubuntussa. VLC:n joustavuus ja luotettavuus tekevät siitä yhden suosituimmista vapaiden ohjelmistojen videotyökaluista.

ASENNUS

VLC on työpöytäsovellus, joka sinun täytyy ladata ja asentaa. Asennuksen askeleet riippuvat käyttämästäsi alustasta (GNU/Linux, Mac OS/X tai Windows).

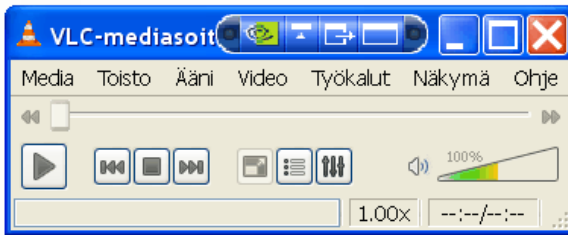
Tällä sivulla on linkit VLC:n lataamiseen, sekä ohjeet eri alustoille:

<http://www.videolan.org/vlc/>

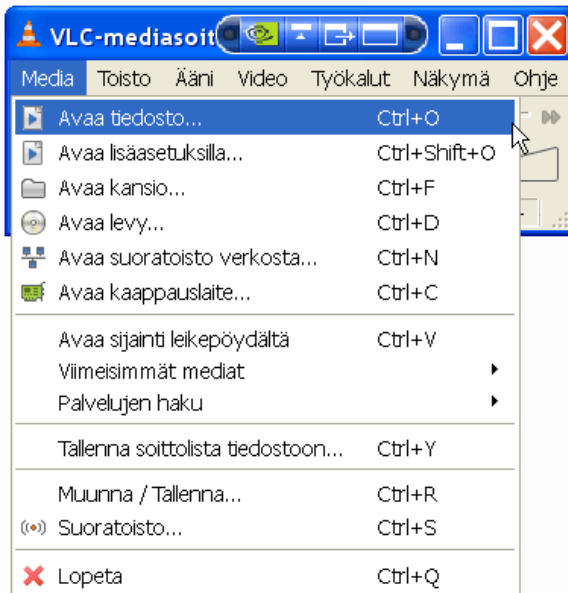
FLOSS Manualsin englanninkielisellä verkkosivulla on myös hyvä opas VLC:n asentamisesta ja käytöstä.

VIDEON TOISTO

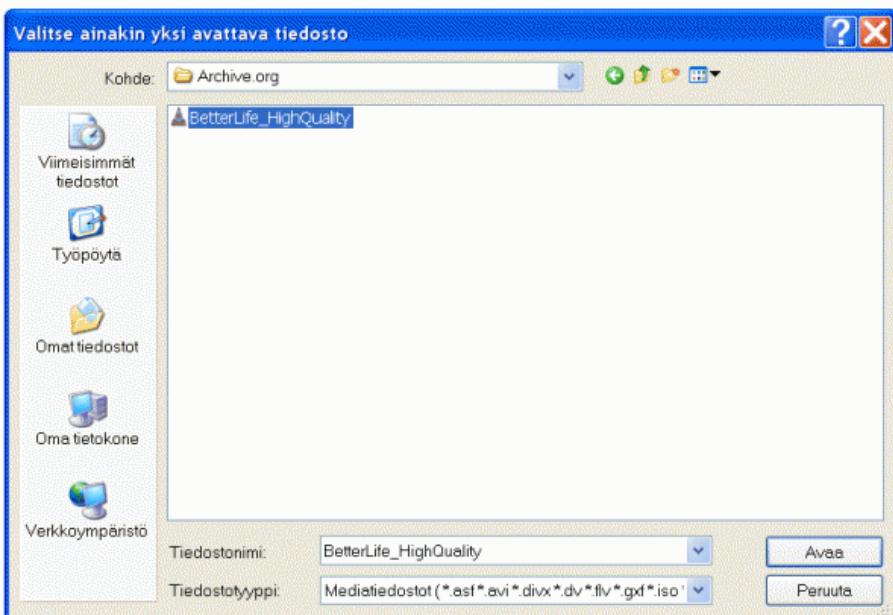
1. Käynnistä VLC. Näet tällaisen ikkunan:



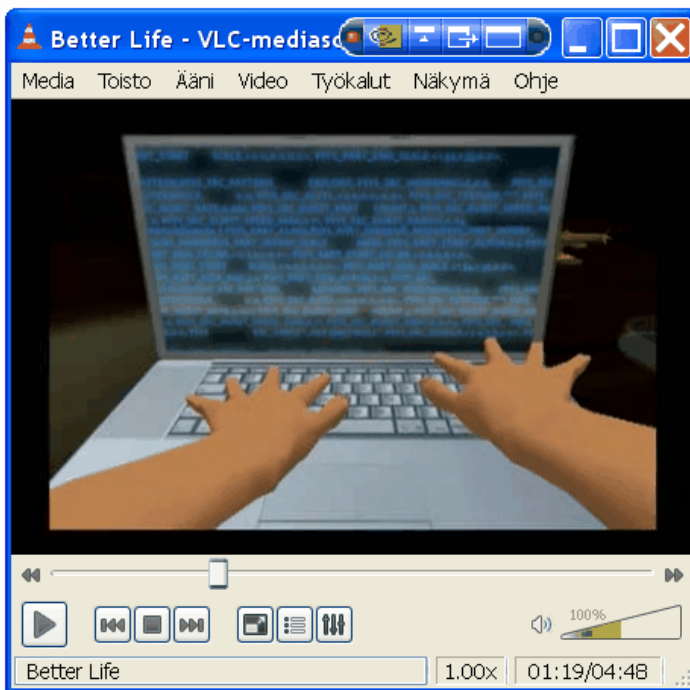
2. Mene VLC:n valikossa kohtaan Media > Avaa tiedosto.



3. Etsi Theora-tiedosto, jonka tahdot avata. Valitse se ja klikkaa Toista.



4. Video ilmestyy tällaiseen ikkunaan ja VLC alkaa toistaa sitä.



TOISTON PERUSOHJAIMET

- Aivan kuin useimmissa mediasoittimissa, voit siirtää toiston sijainnin osoitinta eteenpäin tai taaksepäin videossa.
- Voit kaksoisnapsauttaa videota, jolloin se siirtyy koko ruudun täyttävään tilaan. Kaksoisnapsauta uudestaan, niin pääset pois koko ruudusta.

MIRO

Miro (jota kutsuttiin aiemmin nimellä Democracy TV) on työkalu videon etsimiseen, lataamiseen ja katselemiseen monista verkkolähteistä. Se on vapaa ohjelmisto, jonka on tehnyt kansalaisjärjestö Participatory Culture Foundation: <http://participatoryculture.org/>.

Miro soittaa suuren joukon erilaisia formaatteja, joihin kuuluu Theora, ja se toimii monilla erilaisilla alustoilla (käyttöjärjestelmät, kuten Windows, OS/X ja GNU/Linux). Videon toistamisen lisäksi Mirolla on mahdollista etsiä ja ladata videoita erityisesti formatoiduilta videolistoilta, jotka tunnetaan podcasteina tai vodcasteina.

Voit käyttää Miroa toistamaan videon työpöydälläsi, tai lataamaan ja katsomaan videon URLista, joka osoittaa suoraan videotiedostoon, tai jopa suositulta videosivulta, kuten YouTube. Voit myös käyttää Miroa Bittorrent-tiedostojen lataamiseen ja katseluun (tiedostojen päätte on .torrent).

ASENNUS

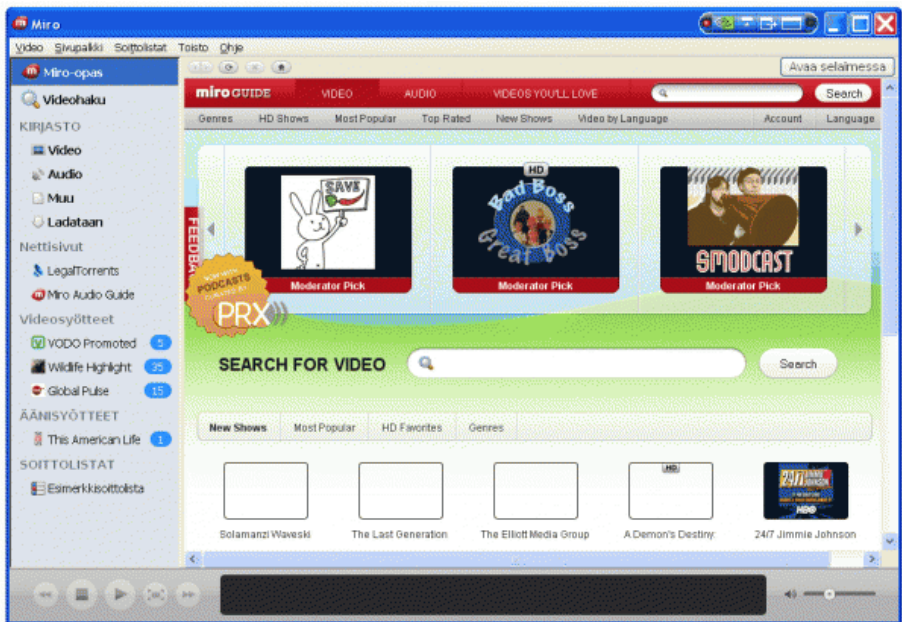
Miro on työpöytäsovellus, joka sinun täytyy ladata ja asentaa. Asentamisen askeleet riippuvat käyttämästäsi alustasta (GNU/Linux, Mac OS/X tai Windows).

Tällä sivulla on latauslinkit Miroon ja ohjeet eri alustoille:

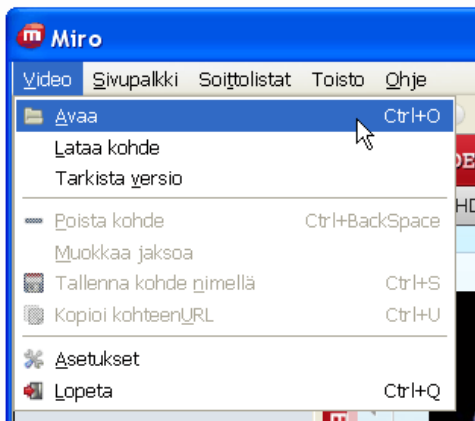
<http://getmiro.com>

VIDEON TOISTO

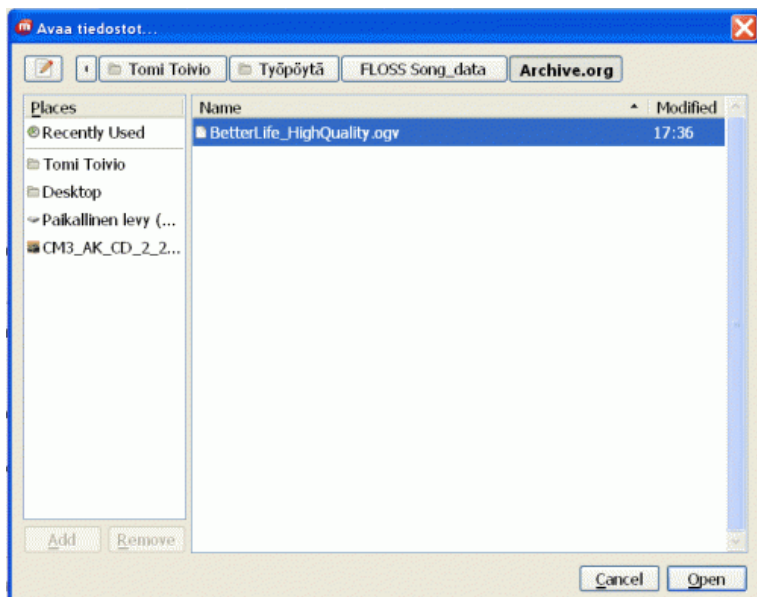
1. Käynnistä Miro. Kun olet napsautellut läpi muutaman viestin, jotka näkyvät Miron toimiessa ensimmäistä kertaa, näet tällaisen ikkunan:



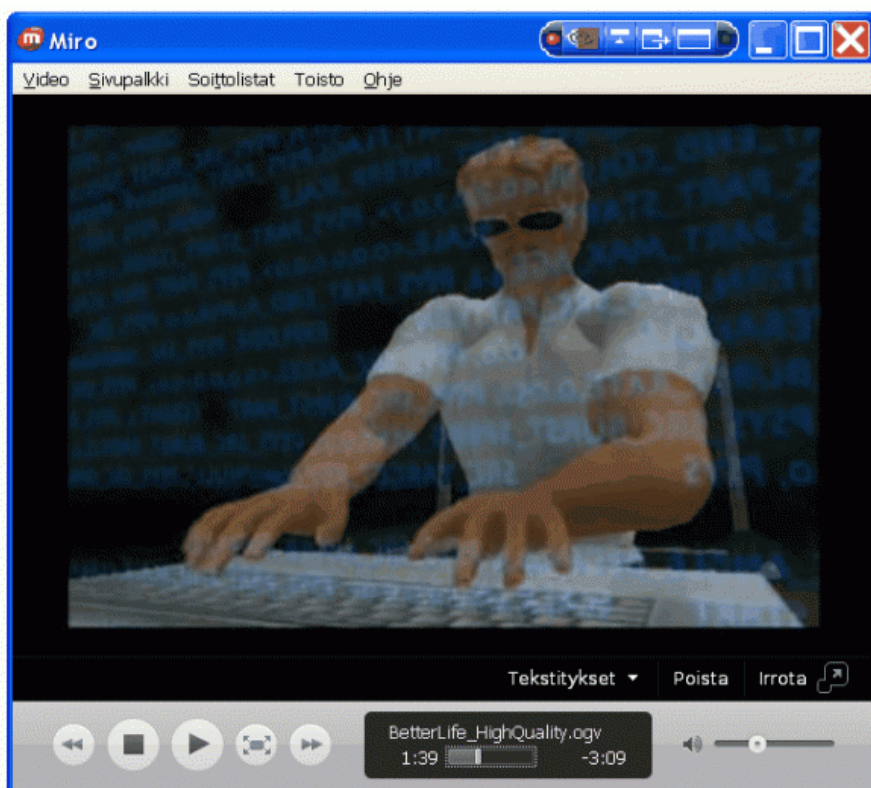
2. Mene Miron hakemistossa kohtaan Video > Avaa.



3. Etsi tiedosto, jonka tahdot avata. Valitse se ja napsauta "Avaa".



4. Videon pitäisi alkaa toistua Miron ikkunassa.



THEORA VERKOSSA

10. JOHDANTO

11. HTML5

12. THEORA SIVUILLASI

13. HOTELLEJA THEORALLE

JOHDANTO

Täytyy harkita muutamaa asiaa ennen videon laittamista verkkoon.

Huomaa, että katsojilla täytyy olla laajakaistayhteys, jotta he voivat katsoa videotasi verkkosivullasi. Ja mitä korkeampi videon laatu on, sitä suurempaa kaistanleveyttä videon katsomiseen tarvitaan. Tämä merkitsee, että joskus kohdeyleisö ei edes näe videota verkossa, ja siksi sinun täytyy harkita toista strategiaa, kuten videon levittämistä CD:llä, DVD:llä tai USB-muistitikuilla.

Jos yleisölläsi on nopea yhteys, tahdot ehkä tarjota videon ladattavaksi tai toistettavaksi verkkosivullasi. Jokaisella strategialla on puolensa riippuen siitä, mitä tahdot saavuttaa.

Edut videon tarjoamisessa toistettavaksi verkkosivulta ovat lähinnä nämä:

1. Voit määritellä videon esittämisen kontekstin.
2. Yleisölle on käytännöllistä vain napsauttaa nähdäkseen videon.
3. Sivujen vierailijoille on näppärää napsauttaa videota ja katsoa siitä pieni osa yleiskuvan saamiseksi, sen sijaan että lataisivat koko tiedoston saadakseen esivaikutelman työstäsi.
4. Yleisölle on helppoa näyttää työtäsi muille, esimerkiksi lähettämällä linkki videoon, jonka he tahtovat näyttää kavereilleen.

Edut videon tarjoamisessa ladattavaksi ovat:

1. Yleisösi voi jakaa sen tiedostonjakoverkoissa (jotkut pitävät tätä haittapuolena)
2. He voivat toistaa sen miten tahansa monta kertaa tahtovat kuluttamatta ylimääräistä kaistanleveyttä.
3. He voivat muokata sitä.
4. He voivat näyttää sen ilman verkkoyhteyttä (esimerkiksi elokuvanäytöksissä).
5. Voit tarjota korkealaatuisemman videon kuin voisit tarjota verkkosivullasi selaimen tai kaisanleveyden rajoitusten vuoksi.

Voit tarjota myös molemmat vaihtoehdot: mahdollisuuden katsoa video verkkosivullasi ja mahdollisuuden ladata video. Voit jopa ladata sivuillesi kaksi versiota: toinen on koodattu verkkoon sopivalla laadulla (hieman kuin esikatseluversio elokuvasta pienemmässä koossa ja huonommalla laadulla) ja toinen video ihmisille ladattavaksi paremmalla laadulla ja suuremmalla resoluutiolla.

Jos tahdot jakaa videosi verkkosivullasi sinulla on muutama vaihtoehto. Ensinnäkin voit käyttää hienoa HTML5 -videotägiä, ja jos tämä ei ole mahdollista jokaiselle katsojalle, voit myös tarjota vaihtoehtoa: Cortado-soitinta. Jos sinulla ei ole pääsyä tarvittavaan palvelininfrastruktuuriin, voit edelleenkin käyttää yhtä monista tarjolla olevista videopalveluista, jotka tarjoavat ilmaista tilaa Theora-videoille (et tahdo ladata videotasi YouTubeen kaltaisille sivuille, sillä videon laatu heikkenisi, ja menettäisit joitain oikeuksistasi materiaaliin).

Toinen strategia on tarjota video ladattavaksi Bittorrentilla. Bittorrent on käyttäjien välinen tiedostonjako-protokolla, joka jakaa kaistanleveyden tiedostoa samaan aikaan lataavien tietokoneiden välillä. Tällöin siirtö muuttuu sitä nopeammaksi mitä useammat ihmiset lataavat tiedostoa. Bittorrent on luultavasti taloudellisin tapa siirtää isoja tiedostoja internetissä.

HTML5-VIDEO

Jos luot videon, tahdot ehkä näyttää sen verkkosivulla. Koodi, jolla verkkosivu luodaan, perustuu sääntöihin, joita kutsutaan HTML:ksi, ja näistä säännöistä on tullut juuri uusi versio, jota kutsutaan HTML5:ksi.

HTML5 sisältää uuden ominaisuuden: videotägin. "Tägi" on muutama rivi HTML-koodia, joka käskää selainta näyttämään jotain tai tekemään jotain. HTML5 videotägi sallii yksinkertaisen videoiden integraation samaan tapaan kuin kuvia liitetään verkkosivulle.

Videota voidaan myös näyttää hyvin käyttökelpoisten ohjaimien kanssa. Ohjaimilla videon voi toistaa, pysäyttää, äänenvoimakkuutta voi vaihtaa, ja videon aikajanaa voi vierittää.



PERUSSYNTAKSI

Tässä on perusesimerkki videon sisällyttämisestä sivulle HTML5-koodin avulla:

```
<video src="video.ogv"></video>
```

Edellinen esimerkki liittää videotiedoston 'video.ogv' verkkosivulle. Tämän esimerkin tiedoston pitäisi olla samassa hakemistossa kuin HTML-tiedoston, koska "src"-parametri viittaa paikalliseen tiedostoon. Jos tahdot viitata toisessa hakemistossa olevaan tiedostoon samalla verkkopalvelimella, joudut tarjoamaan polkutiedot aivan kuin tekisit kuvatiedoston tapauksessa.

```
<video src="/bin/edit/myvideofiles/video.ogv"></video>
```

Voit myös määritellä toisella palvelimella olevan tiedoston:

```
<video src="http://mysite.com/video.ogv"></video>
```

PARAMETRIT

Lisäparametrien lisääminen antaa lisää hallintaa videon suhteen.

```
<video tabindex="0" src="video.ogv">< height="320" width="480">
```

Tässä esimerkissä videon korkeus (height) ja leveys (width) on määritelty. Jos et tahdo kuvan näyttävän vääristyneeltä, on tärkeää asettaa korkeus- ja leveysulottuvuudet oikein. autoplay merkitsee, että video soitetaan automaattisesti sivun latauduttua. controls merkitsee, että videon toistamiseen ja pysäyttämiseen liittyvät ohjaimet ovat näkyvillä.

On mahdollista liittää tekstiä tai muuta HTML-sisältöä videotägiin vaihtoehtoisena sisältönä, joka näytetään, jos selain ei tue videotägiä.

OMIEN OHJAINTESI TAI SOITTIMEN KUOREN KÄYTTÖ

Jos osaat hieman JavaScriptiä, voit hallita helposti videon toistoa. Sen sijaan, että käyttäisit selaimen tarjoamia ohjaimia, voit luoda oman käyttöliittymän ja ohjilla videoelementtejä JavaScriptin avulla. On kaksi asiaa, jotka sinun täytyy muistaa tätä metodia käyttäessäsi:

1. Älä unohda pudottaa controls-attribuuttia
2. Video tarvitsee id-parametrin, kuten tässä:

```
<video src="video.ogv" id="omavideo"></video>
```

Joitain JavaScript-funktioita:

```
video = document.getElementById("omavideo");  
//toista video  
video.play();  
//pysäytä video  
video.pause();  
//mene 10 sekunnin kohdalle videossa  
video.currentTime = 10;
```

Jos sinulla on monta videotägiä yhdellä verkkosivulla, sinun tulee antaa jokaiselle yksilöllinen id, jotta JavaScript tietää, mihin videoon ohjaimet viittaavat.

Videotägien tarjoamien funktioiden ja tapahtumien koko lista löytyy HTML5-spesifikaatiosta sivulla <http://www.whatwg.org/specs/web-apps/current-work/#video>

MANUAALISET VAIHTOEHDOT

Edellisessä yksinkertaisessa esimerkissä videoelementti näyttää tekstin videoelementin sisällä, jos selain ei tue sitä.

Tekstin sijasta on mahdollista käyttää Cortadoa vaihtoehtona, jos selain tukee Javaa. Cortado on avoimen lähdekoodin useamman alustan ja selaimen Ogg Theora -soitin, joka on kirjoitettu Javalla. Hieno asia on, että käyttäjän ei tarvitse ladata mitään ylimääräisiä Java-paketteja, koska sovelma käyttää selaimen sisäistä Javaa. Cortadon kotisivu löytyy täältä:

<http://www.theora.org/cortado/>

Esikäännetty sovelman versio on myös saatavilla tästä osoitteesta:

<http://www.theora.org/cortado.jar>

Voit ladata jar-tiedoston tai voit viitata siihen suoraan. Seuraavassa esimerkissä Cortado liitetään sivulle (kaikkia parametrejä ei vaadita, mutta tämä lista antaa kuvan mahdollisista vaihtoehtoista):

```
<applet code="com.fluendo.player.Cortado.class"  
        archive="http://www.theora.org/cortado.jar"  
width="352" height="288">  
  <param name="url" value="http://myserver.com/theora.ogv"/>  
  <param name="framerate" value="29"/>  
  <param name="keepAspect" value="true"/>  
  <param name="video" value="true"/>  
  <param name="audio" value="true"/>  
  <param name="bufferSize" value="100"/>  
  <param name="userId" value="user"/>  
  <param name="password" value="test"/>  
</applet>
```

Jos tahdot ladata jar-tiedoston vaihtoehtona, varmista että laitat sen (**cortado.jar**) ja edellisen HTML-sivun samaan hakemistoon. Vaihda sitten seuraava rivi sisältämään viite (linkki) omaan OGG-strimiin (suoraan lähetykseen tai esinauhoitettuun):

```
<param name="url" value="http://palvelimeni.fi/theora.ogv"/>
```

Jos nyt avaat verkkosivun selaimessasi, se soittaa videon.

Käyttääksesi Cortadoa vaihtoehtona, aseta Cortado-tägi HTML5-videotägin sisään - kuten seuraavassa esimerkissä:

```
<video src="video.ogv" width="352" height="288">
  <applet code="com.fluendo.player.Cortado.class"
    archive="http://theora.org/cortado.jar" width="352" height="288">
    <param name="url" value="video.ogv"/>
  </applet>
</video>
```

JAVASCRIPTIIN PERUSTUVAT SOITTIMET

On olemassa JavaScript-kirjastoja, jotka hoitelevat vaihtoehdon valinnan. Nämä kirjastot mahdollistavat yksinkertaisen sivulle liittämisen, vaikka säilyttävät vaihtoehtoiset soittimet ja toistomenetelmät eri selaimissa.

Mv_Embed

Mv_embed -kirjastoa on helppo käyttää. Yksi JavaScript-tiedosto mahdollistaa HTML5-videotägin käytön ja ominaisuuksien uudelleenkirjoittamisen soittimelle, joka toimii monissa selaimissa ja lisäosissa. [Lisätietoa aiheesta mv_embed](#)

```
<script type="text/javascript" src="http://metavid.org/w/mwScriptLoader.php?
class=mv_embed"></script>
...
<video src="elokuvani.ogv" poster="elokuvani.jpeg">
```

ltheora

[lTheora](#) on PHP-skripti, joka mahdollistaa ogg/theora/vorbis-videon (ja audion) toistamisen. Se on helposti asennettava ja käytettävä. ltheoran sivulla on dokumentaatiota soittimen ja kuorien käytöstä.

TUKI SELAIMISSA

Juuri nyt Firefox 3.5 on ainoa julkaistu selain, joka tukee Theoraa. Opera ja Google Chrome tarjoavat Theora-tuen beta-versioissaan. Safari tukee videotägiä, mutta tukee koodekkeja vain QuickTimen kautta - tämä merkitsee, että se ei tue Theoraa. XiphQT (<http://xiph.org/quicktime>) lisää Theora-tuen QuickTimeen ja Safariin.

VIDEOT OMALLA SIVUSTOLLA

Voit laittaa Ogg Theora -videot omalle verkkopalvelimelle, aivan kuin kuvat tai HTML-tiedostot.

MIME-TYYPIT

Jotta videot toimivat niillä täytyy olla oikea **mime-tyyppi**.

Mime-tyyppi merkitsee tapaa erilaisten internetissä välitettyjen tiedostotyyppien tunnistamiseen. Tämä informaatio välitetään yleensä datan mukana. Lisätieto identifioi millaista informaatiota välitetään ja se ei yleensä ole ihmisten luettavissa, mutta ohjelmisto tulkitsee sen siten, että oikeanlainen data välitetään ja sen prosessoivat oikeanlaiset ohjelmat. Informaatio lähetetään siirretyn datan otsikossa.

Otsikko on pieni määrä metadataa, joka lähetetään ohjelmasta toiseen, ja joka kuvailee siirretyn tiedon tyyppiä. Tyyppilinen otsikko sisältää pituuden, määränpään, mime-tyypin jne.

Mime-tyypeissä on kaksi osaa, tyyppi ja alatyyppi (joskin näihin kahteen viitataan vain "tyyppinä"). Tyyppi kirjoitetaan muodossa tyyppi/alatyyppi. On vain neljä tyyppikategoriaa - audio, video, teksti ja sovellus. Alatyyppejä on lukemattomia.

Oikea mime-tyyppi Theora-videolle on "video/theora".

Nykyisen palvelimen tulisi lähettää oikeat tiedot. Jos palvelimesi ei lähetä oikeita otsikoita, sinun täytyy muuttaa verkkopalvelimesi asetuksia. Jos käytät Apache-verkkopalvelinta (verkon suosituin palvelin) on kaksi tapaa tehdä tämä:

1. voit lisätä kaksi riviä Apache-verkkopalvelimen asetuksiin
2. voit mahdollisesti tarjota lisäasetuksen laittamalla videokansioosi .htaccess -tiedoston

Ensimmäiseen strategiaan tarvitset pääsyn verkkopalvelimesi konfiguraatiotiedostoon (httpd.conf). Tässä tapauksessa kaikki verkkopalvelimen välittämät tiedostot lähettävät oikean tiedon. Tämä on paras ratkaisu. Kuitenkaan sinulla ei ole pääsyä verkkopalvelimen konfiguraatiotiedostoon (esimerkiksi, jos käytät jaettua verkkopalvelinpalvelua) ja tahdot ehkä kokeilla toista strategiaa. Toinen strategia vaikuttaa vain videoon, joka tarjotaan samasta hakemistosta, johon laitoit .htaccess-tiedoston.

Molempia strategioita varten tarvitset informaation asiaankuuluvasta tiedostosta:

```
AddType video/ogg          .ogg
AddType application/ogg     .ogg
```

Tiedostoille httpd.conf ja .htaccess voit sijoittaa tämän tiedon tiedoston loppuun. Jos sinulla ei ole .htaccess-tiedostoa, voit vain luoda tyhjän tiedoston ja lisätä tämän tiedon (muuta tietoa ei tarvita).

OGGZ-CHOP

Apachea käyttämällä voit myös ryhtyä paljon hienostuneemmaksi. Voit esimerkiksi mahdollistaa vain tiettyyn Theora-videon osaan viittaavien ja sen toistavien URLien käytön. Jos tahdot liittää vain osan videostasi verkkosivuusi tai sallia linkittämisen tiettyyn ajankohtaan videossa, voit käyttää oggz-chop -pakettia serverilläsi. Oggz-chop asennettuna voit käyttää osoitteena tiettyä videon osaa laittamalla siirros-parametrisin urliin. Käyttäisit tätä näyttääksesi sekunnista 23 sekuntiin 42:

```
<video src="http://example.com/video.ogv?t=23.0/42.0"></video>
```

oggz-chopin asentaminen vaatii toimintamoduulin sallimista Apachessa ja oggz-chopin asentamista palvelimelle. Tämän toimintamoduulin lisääminen on tämän dokumentin laajuuden ulkopuolella, joten sinun on paras ottaa yhteyttä Apache-guruun tai lukea aiheesta dokumentaatiota ennen kuin yrität tätä.

Kuitenkin... Jos sinulla on Apache2 asennettuna ja ylläpitäjän tunnus palvelimellesi, voit kokeilla tätä komentoa asentaaksesi toimintamoduulit:

```
sudo a2enmod actions
```

Yrität sitten omalla vastuullasi...

oggz-chop on osa oggz-tools -pakettia, voit asentaa sen komennolla:

```
sudo apt-get install oggz-tools
```

Kun nuo on asennettu, sinun täytyy sallia oggz-chopin käyttö näillä kahdella rivillä Apachen konfiguraatiotiedostossa tai .htaccess-tiedostossa:

```
ScriptAlias /oggz-chop /usr/local/bin/oggz-chop
Action video/ogg /oggz-chop
```

ETÄKÄYTÖN SALLIMINEN

Pelkästään kuvia voidaan laittaa verkkosivuille muilta sivustoilta, ei videoita, elleivät nuo sivustot nimenomaan salli tätä. Salliaksesi muiden sivustojen videoidesi käytön domainissaan, lisää tämä rivi Apachen konfiguraatio tai .htaccess -tiedostoon:

```
Header Set Access-Control-Allow-Origin "*"
```

Tällä asetuksella palvelin vastaa lisäotsikolla 'Access-Control-Allow-Origin: *', joka tarkoittaa, että videot voidaan liittää sivuille missä tahansa domainissa sivulta sivulle. Jos tahdot rajoittaa pääsyn videoihin sallituksi ainoastaan osoitteesta <http://example.org>, joudut muuttamaan sen tällaiseksi:

```
Header Set Access-Control-Allow-Origin "http://example.org"
```

Huomaa, että nyt videota ei voida laittaa domaineihin, jotka ovat muita kuin example.org. Access-Control-Allow-Origin -otsikko voi myös sisältää pilkulla erotellun listan sallituista domaineista.

https://developer.mozilla.org/En/HTTP_access_control sisältää yksityiskohtaisemman kuvauksen http-pääsynvalvonnasta.

VIDEOIDEN TARJOAMINEN SKRIPTILLÄ

Vielä yksi vaihtoehto. Jos sinulla ei ole mahdollisuutta vaikuttaa palvelimesi asetuksiin, mutta tahdot silti käyttää videoita, voit käyttää pientä PHP tai CGI -skriptiä asettaaksesi oikeat otsikot ja tarjotaksesi videon. Sellainen skripti voisi näyttää tältä:

```
<?php
$video = basename($_GET['name']);
if (file_exists($video)) {
    $fp = fopen($video, 'rb');
    header('Access-Control-Allow-Origin: *');
    header('Content-Type: video/ogg');
    header('Content-Length: ' . filesize($video));
    fpassthru($fp);
} else {
    echo "404 - video not found";
}
?>
```

Jos skripti on sijoitettu nimellä index.php videohakemistoosi, voisit käyttää osoitetta <http://example.com/videos/?name=test.ogv> sen sijaan että linkität suoraan videoosi (<http://example.com/videos/test.ogv>).

<video src="http://example.com/videos/?name=test.ogv"></video>

VERKKOHOTELLIT

Monet verkkosivut tarjoavat ilmaisia verkkohotellipalveluja OGG Theora -sisällöllesi. Voit linkittää niihin suoraan verkkosivuiltasi, tai antaa ihmisille linkin ulkoisella sivustolla olevaan sisältöön.

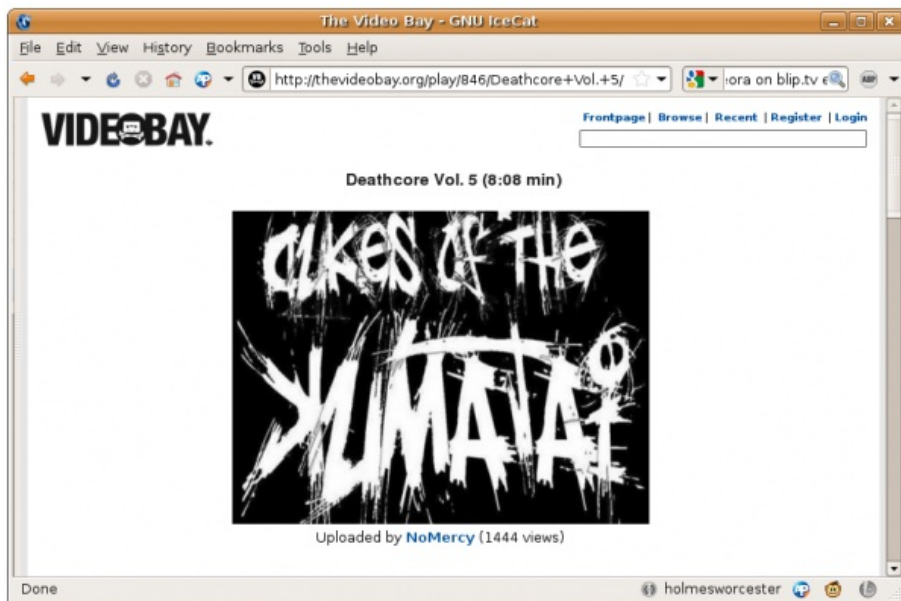
TINYVID.TV

TinyVid (<http://tinyvid.tv/>) on yksinkertainen videonjakopalvelu, jonka Theora-yhteisön jäsenet ovat kehittäneet. Voit ladata videon missä tahansa formaatissa, ja se koodataan Theora-muotoon. Se tukee myös Firefogg-laajennusta Firefoxiin (<http://firefogg.org/>), jolloin voit koodata ja ladata videon sivustolle samaan aikaan!



THE VIDEO BAY ([THEVIDEOBAY.ORG](http://thevideobay.org/))

Video Bay on The Pirate Bayn (<http://thepiratebay.org/>) ylläpitäjien uusi projekti. Tämä sivusto koodaa videon mistä tahansa formaatista Theoraan. Sivuston ylläpitäjillä on maine ihmisten etujen puolustajina tekijänoikeuksien hallitsijoita vastaan.



ARCHIVE.ORG

Archive.org on verkkosivusto, jota ylläpitää voittoa tuottamaton järjestö **Internet Archive**. Järjestö keskittyy digitaalisen median säilyttämiseen. Lisätäkseen videon arkistoon, sinun täytyy lisensoida työ **Creative Commons** (tai vastaavalla) lisenssillä. Tämä ei yleensä ole ongelma, jos teit teoksen itse, mutta voi olla ongelma, jos käytit tekijänoikeuden alaista materiaalia (esimerkiksi musiikkia videon sisällä) työssäsi, tai jos lataat jotain jonkun muun tekemää.

Voit tarjota Ogg Theora -sisältöä Internet Archivessa ilmaiseksi, ja lisätä siihen linkin omalta verkkosivultasi. Archive.org muuntaa videosi Ogg Theora -muotoon ja moniin muihin formaatteihin, kun lataat sen arkistoon.

Huomaa: kun käyt sivulla Archive.org, se huomauttaa jos selaimesi tukee videotägiä. Jos näet tämän viestin...



Lataa yksinkertaisesti sivu uudelleen ja katso viestiä videoikonin alla:



[embed this](#)

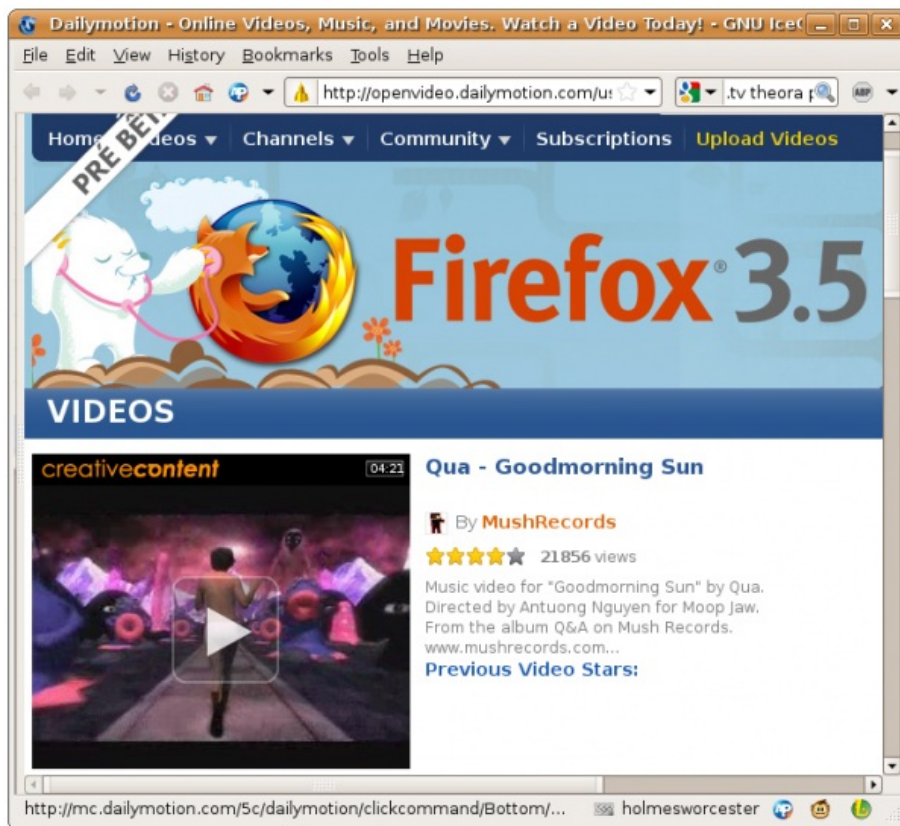
Your browser supports the new <video> tag!
Would you like to [try the new <video> tag?](#)

Tässä vaiheessa klikkaa linkkiä, joka sanoo "try the new tag" ja klikkaa seuraavaa linkkiä käyttääksesi aina videotägiä. Tämän jälkeen videosi näkyvät Theorana. Jos tahdot suoran linkin videon Theora-version, kopioi "Ogg Video" -linkki vasemmassa sivupalkissa. Kun sinulla on tämä linkki, voit linkata suoraan videoon, ja kuka tahansa yhteensopivalla selaimella varustettu näkee videon toiston:



DAILYMOTION

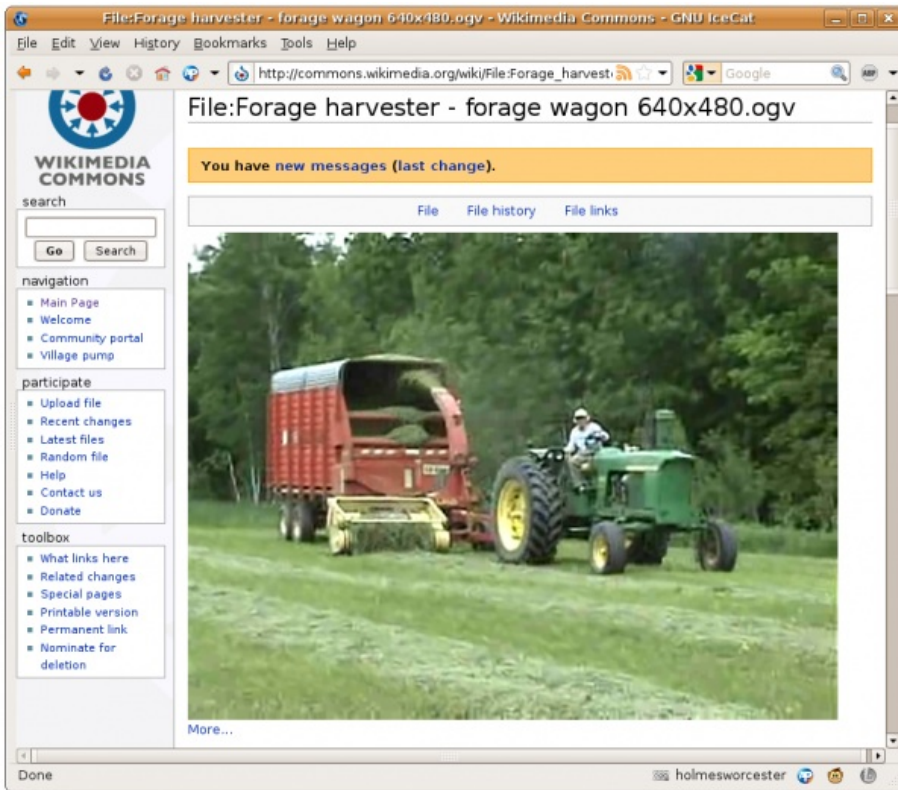
Dailymotion (<http://dailymotion.com/>) on yksi isommista videonjakosivustoista, ja se on näperrellyt Theora-tuen kanssa viime aikoina. 300 000 videota on muunnettu Theoraan, ja anomalla "motion maker" -käyttäjätiliä (ja saamalla hakemuksen hyväksytyttyä) voit julkaista siellä videoita käyttäen Theoraa. Heidän kokeellinen Theora-portaalinsa on täällä: <http://openvideo.dailymotion.org/>. Huomaa: kaikki videot tässä portaalissa eivät näky vielä Theorana, joten tarkasta että videosi toimivat ennen kuin luotat tähän palveluun.



WIKIMEDIA COMMONS

Wikimedia Commons (<http://commons.wikimedia.org/>) on verkkosivu, jota ylläpitää *Wikimedia Foundation* (<http://wikimediafoundation.org/>), voittoa tuottamaton organisaatio, joka ylläpitää myös Wikipediaa. Se on tietokanta mediatiedostoista, jotka ovat saatavilla kenelle tahansa käytettäväksi mihin tahansa tarkoitukseen. Se on avoin verkkosivu, johon kuka tahansa voi tarjota sisältöä, ja joka käyttää wiki-ohjelmistoa, joka mahdollistaa helpon yhteistyön.

Tätä sivustoa hallinnoivat täysin vapaaehtoiset toimittajat, jotka luovat myös suurimman osan sisällöstä tarjoamalla omia töitään. Yhteisö on monikielinen, kääntäjiä on tarjolla tusinoille kielille. Se kerää vain materiaalia, joka on tarjolla vapaiden sisältölisenssien alla tai kuuluu public domainiin.



- ENKODDAUS
14. JOHDANTO
15. FIREFOGG
16. VLC
17. FFMPEG2Theora
18. THOGGEN

ENKODDAUS

Koodaus merkitsee prosessia, jossa Theora-tiedosto luodaan pakkaamattomasta videon raakamateriaalista. Tässä tapauksessa lähdemateriaali on jossain pakatussa muodossa, väliin tulee koodauksen askel, jota tarvitaan ennen Theora-videon luomista. Tämä koodaus-dekoodaus tunnetaan usein nimellä transkoodaus, tosin tässä kirjassa käytämme synonyyminä termiä koodaus.

Koodauksen toteuttavia ohjelmia kutsutaan koodaajiksi. Monia Theora-koodaajia on olemassa, esimerkiksi ffmpeg2theora ja VLC, nimitäksemme muutamia.

Ennen koodausta käyttäjän täytyy päättää ainakin kaksi parametria:

- luodun Theora-tiedoston kuvan laatu
- äänen laatu

Käytetystä koodaajasta riippuen enemmän vaihtoehtoja voi olla tarjolla koodausprosessin kontrolloimiseksi:

- kuvaruutujen reunojen leikkaus määritellyn määrän mukaisesti koodauksen aikana
- videon resoluution uudelleenskaalaus
- videon kuvataajuuden muuttaminen
- videon ja äänen synkronisaation käsittely
- avainruutujen välin asettaminen

VIDEON LAATU, BITTINOPEUS JA TIEDOSTON KOKO

Useimmat Theora-koodaajat antavat käyttäjän määritellä suoraan koodatun videon subjektiivisen laadun, yleensä skaalalla 0-10. Mitä korkeampi laatu, sitä suurempia ovat tuloksena syntyvät Theora-tiedostot. Useimmat koodaajat voidaan vaihtoehtoisesti konfiguroida koodaamaan annettua keskimääräistä bittinopeutta varten. Vaikka tämä vaihtoehto on käyttökelpoinen Theora-videotiedostojen luomiseen striimausta varten, se tuottaa alle optimaalista laatua ja sitä pitäisi välttää.

Uusimmat versiot joistain Theora-koodaajista sisältävät 2-pass koodauksen. 2-pass koodaus antaa koodaajan asettaa määritellyn kohdebittinopeuden, jossa on optimaalinen videolaatu, ja sen pitäisi siten olla samassa tasossa laatuvalvotun koodauksen kanssa, mutta ottaa kaksinkertaisen koodausajan. 2-pass -koodauksella luodut videot eivät ehkä sovi striimaukseen.

VIDEON RESOLUUTIO JA KUVATAAJUUS

On monta syytä, joiden vuoksi luomasi Theora-videon resoluution tulis olla lähdevideota pienempi:

- Jos koodaaja tuottaa liian suuria tiedostoja, jopa matalalaatuisilla asetuksilla nollan ympärillä
- Jos vaadittu tiedoston maksimikoko vaatii hyvin matalia laatuasetuksia (0-3), mikä johtaa liian huonoon näkyvään laatuun
- Jos koodatun videon toiston tulisi toimia jopa matalatehoisilla tietokoneilla
- Jos lähdevideomateriaalisi resoluutio on suurempi kuin perustarkkuuden videoilla; Theoran videokoodekkia ei ole suunniteltu korkean tarkkuuden videoon eikä tee sitä kovin hyvin.

Jos lähdevideomateriaalisi resoluutio on epätavallisen matala, voit säästää bitit, videon resoluution nostaminen koodauksen aikana voi vaikuttaa positiivisesti videon yleiseen laatuun.

Kuvataajuuden säätäminen koodauksen aikana on yleensä huono idea, se johtaa nykimiseen ja madaltaa havaittavaa laatua paljon. Jos kuitenkin tahdot hyvin matalan kohdetiedoston laadun, yritä vähentää kuvataajuutta täsmälleen puoleen lähdetiedoston kuvataajuudesta. Tällöin tiedostokoko voi pienentyä tarpeeksi pudottamatta videolaatua liian alhaiseksi.

VIDEORUUTUJEN KLIPPAUS

Joskus videon lähdemateriaali ei käytä kokonaisia videoruutuja, kuten mustien rajojen jättämistä videon ympärille. On hyvä idea poistaa mustat tai muuten käyttämättömät osat videosta, koska tämä yleensä parantaa koodatun Theora -tiedoston laatua ja tiedostokokoa. Jos mahdollista, yritä pitää videon leveys ja korkeus 16:sta kertoimina. Theora-formaatti kykenee varastoimaan satunnaisten ruutukokojen videota muttei ole siinä kovinkaan hyvä.

VIDEON JA ÄÄNEN SYNKRONISAATIO

Täydellisessä maailmassa koodaaja kopioisi lähdevideomateriaalin videon ja äänen synkronisaation luotuun Theora-tiedostoon. Käytännössä tämä ei vain ole aina mahdollista. Theora-videotiedostojen täytyy pysyä kiinni jatkuvassa ruudussa koko tiedoston läpi. Ääniraitojen äänentoiston nopeus ei myöskään ole jatkuva Theorassa. Jotkin lähdevideoiden kuvataajuus voi kuitenkin vaihdella. Joskus ruutuja puuttuu lähdevideosta nauhoitusvirheiden vuoksi tai johtuen videoleikkausohjelmien käytöstä.

Näissä tapauksissa koodausprosessin täytyy aktiivisesti säätää äänen ja videon synkronisaatiota. Tämä tehdään joko kahdentamalla tai pudottamalla ruutuja videosta, tai muuttamalla audioraitojen nopeutta.

AVAINRUUTUJEN VÄLI

Monet Theora-koodaajat sallivat avainruutujen väliksi kutsutun parametrin muuttamisen. Suurempi avainruutujen väli kutistaa kohdetiedoston kokoa uhraamatta laatua. Avainruudut ovat ne ruudut videossa, joihin soitin voi hakeutua suoraan toiston aikana. Jos videossa on 24 ruutua sekunnissa, avainruutujen väli 240 merkitsee, että etsiminen on mahdollista vain 10 sekunnin jyvityksellä. Leikkauksen ja liittämisen jälkeen koodattu video on rajoittunut avainruutujen jyvitykseen.

Peukalosääntönä ei saa koskaan asettaa avainruutujen väliä yli kymmenen kertaa kohdevideon kuvataajuutta suuremmaksi.

FIREFOGG

Firefogg on Firefox-verkkoselaimen laajennuksen nimi. Firefogg lisää tuen videoiden koodaamiseksi Theoraan käyttäen miellyttävää verkkokäyttöliittymää. Se myös mahdollistaa verkkosivujen käytön videonlatauspalveluun, joka ottaa videoita tietokoneeltasi, kääntää ne lennossa Theora-muotoon ja lataa luodun Theora-tiedoston verkkosivulle.

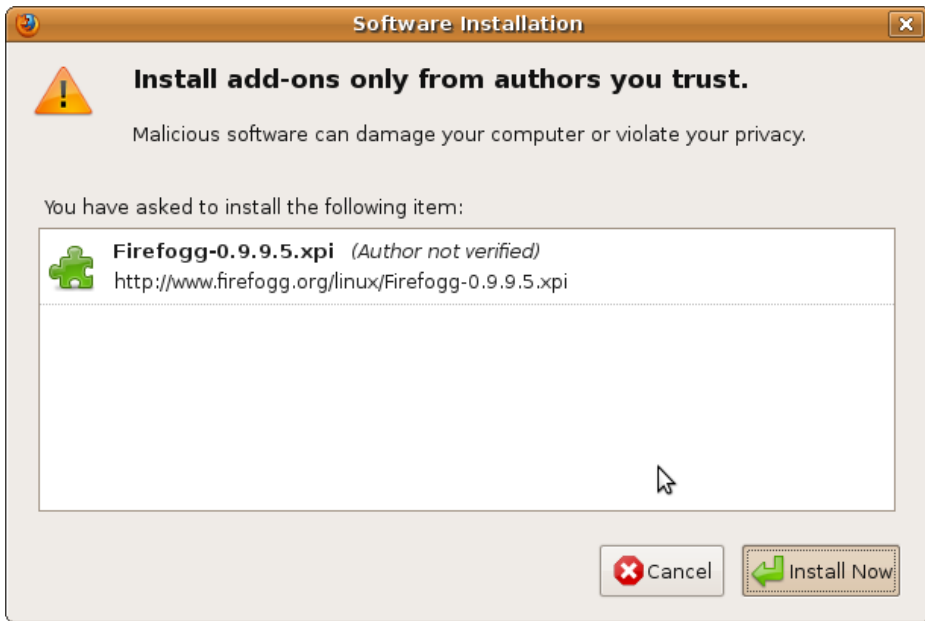
ASENNUS

Firefogg vaatii Firefox-verkkoselainta, vähintään versiota 3.5. Jos Firefox-versiosi on vanhampi, tai jos sinulla ei ole lainkaan Firefoxia asennettuna, vieraile sivulla www.mozilla.com ladataksesi ajan tasalla olevan version.

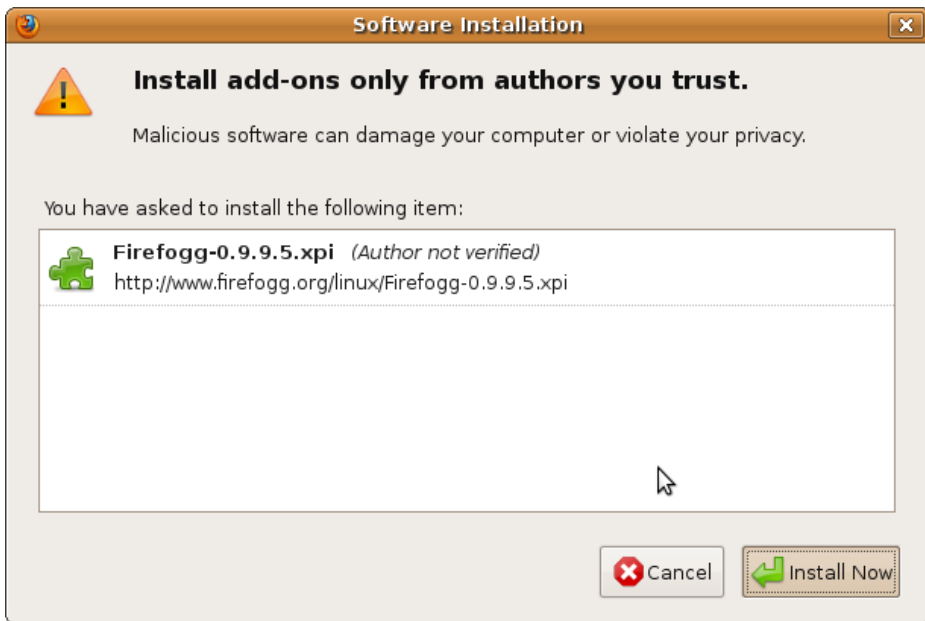
Kun sinulla on viimeisin Firefoxin versio, käytä sitä vieraillaksesi Firefoggin kotisivulla www.firefogg.org:



Nyt voit klikata *Install Firefogg*. Sivun yläosassa Firefox pyytää sinua sallimaan uuden tiedoston asentamisen:



Klikkaa *Allow*, jolloin seuraava dialogi ilmestyy:



Klikkaa *Install Now*. Asennuksen jälkeen sinua pyydetään käynnistämään Firefox uudelleen. Klikkaa *Restart Firefox* päästäksesi eteenpäin.

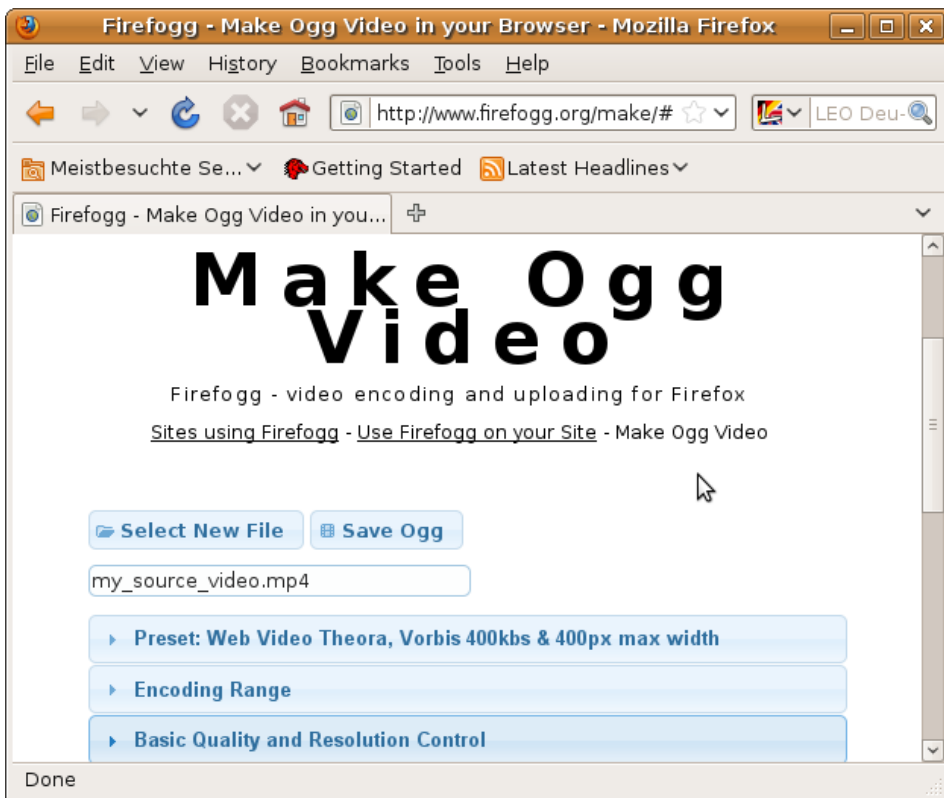
ENSIMMÄISEN VIDEOSI KOODAAMINEN

Koodataksesi videoita Firefoggilla tarvitset internet-yhteyden. Osia koodausohjelmasta on internetissä ja ne asennetaan tietokoneellesi.

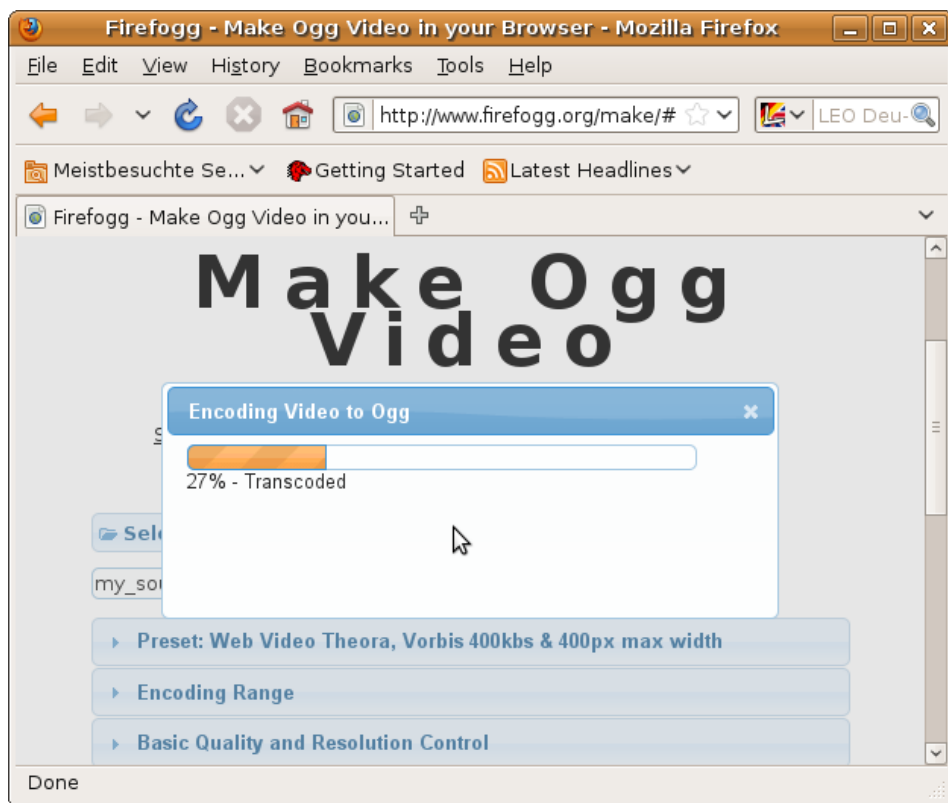
Koodausohjelma käynnistetään käymällä osoitteessa <http://firefogg.org/make>. Seuraava sivusto tulee näkyviin:



Klikkaa *Select File*, jolloin esiin ilmestyy tiedostodialogi, joka antaa sinun valita lähdetiedosto koodattavaksi. Tässä esimerkissä me käytämme videota *my_source_video.mp4*. Tiedoston valitsemisen jälkeen sinut tuodaan dialogiin, joka kysyy videon koodauksen parametrejä:



Valitse tällä kertaa vain *Save Ogg*. Sinua pyydetään valitsemaan luotavan Theora-tiedoston nimi. Muista. Me valitsimme nimen *my_theora_video.ogv* (muista, oikea tiedostonimen päätte Theora-videoille on .ogv). Nyt koodaus alkaa, käyttäen oletusarvoisia parametrejä:

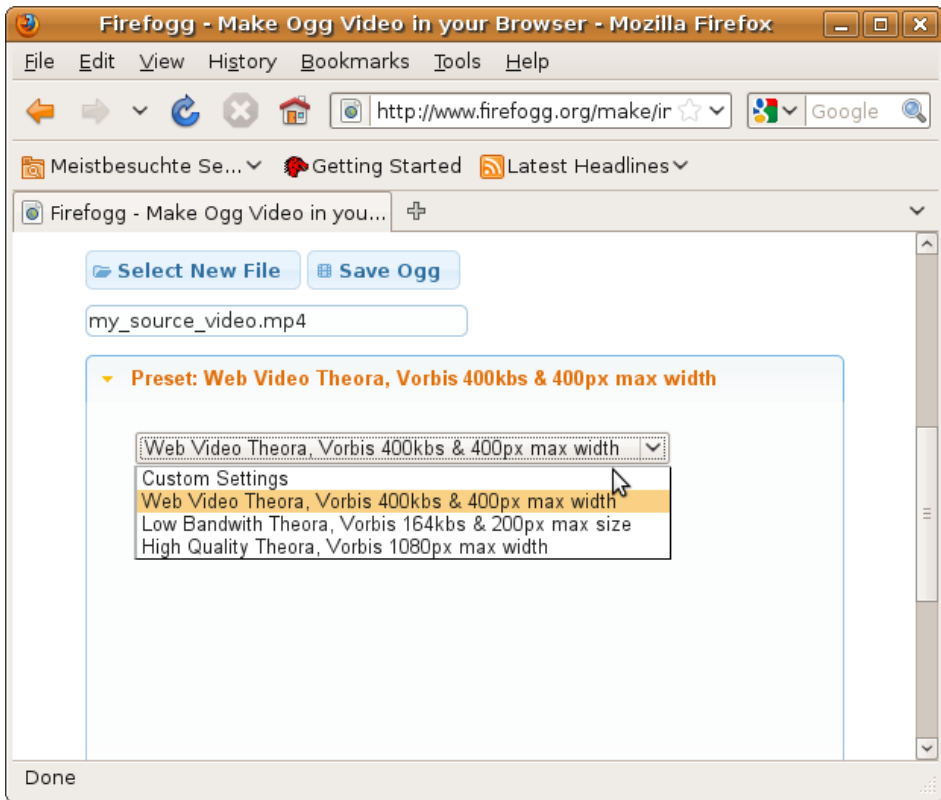


Nyt täytyy odottaa koodausprosessin 100% valmistumista ja sitten se on valmis.

KEHITTYNEEMMÄT KOODAUSOPTIOT

Käyttäen oletusarvoista parametrien joukkoa videon koodaukseen tuottaa hyvin pieniä Theora-tiedostoja, jotka on optimoitu videon striimaukseen verkossa. Videon näkyvä laatu on itse asiassa aika matala useimpien makuun. Mutta odota, Firefogg on yhtä kehittynyt kuin useimmat muut Theora-koodaajat. Pienellä säätämisellä voidaan luoda helposti korkealaatuisia videoita. Säätövaihtoehdot ovat näkyvillä verkkosivulla *Save Ogg* -nappulan alla.

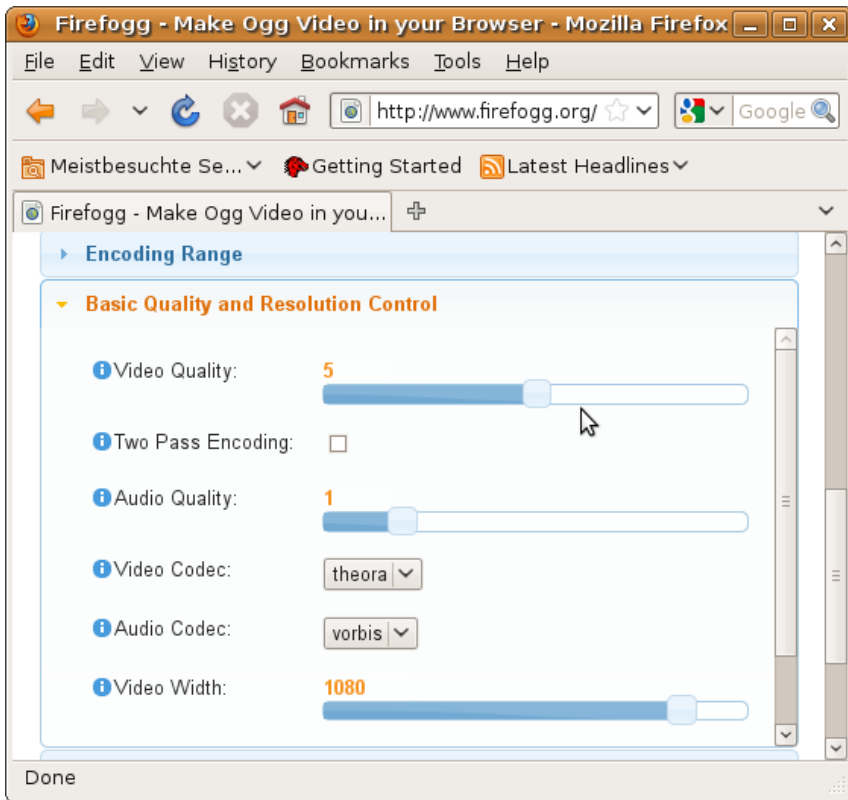
Helpoin tapa koodauksen säätämiseen on valita yksi kolmesta ennakoasetusvaihtoehdosta Preset-valikosta:



Kahta ensimmäistä ennakkovalintaa kutsutaan nimillä *Web Video Theora* ja *Low Bandwidth Theora*. Molemmat on optimoitu videon striimaukseen internetissä. Jos aiot toistaa luotuja Theora-tiedostoja ainoastaan CD:ltä, USB-tikuilta tai tietokoneesi kovalevyltä, sinun ei pitäisi käyttää niitä. Käytä ennakoasetusta *High Quality Theora* niiden sijaan.

Omat asetukset

Useampia koodausprosessin yksityiskohtia voidaan konfiguroida valitsemalla *Custom Settings* ja säätämällä alemman tason optioita muissa tarjolla olevissa valikoissa. Katsotaanpa tärkeintä valikkoa, *Basic Quality and Resolution Control*:



Tässä voit määritellä luodun Theora-tiedoston laadun. Voit myös muuttaa koodatun videon kuvaruudun kokoa. Videon koodaaminen kohdelaatuun (kohdebittinopeuden sijasta) on suositeltu koodaustila Theoraa varten, joten yleensä et tahdo kokeilla muita koodausvaihtoehtoja muissa valikoissa.

MUITA TAPOJA KÄYTTÄÄ FIREFOGGIA

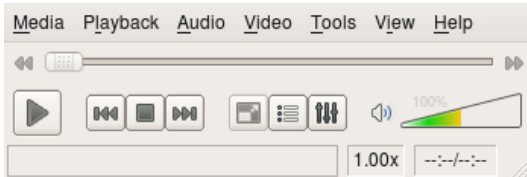
Kun Firefogg on asennettu, sitä voidaan käyttää muilla sitä tukevilla verkkosivuilla. Lista näistä sivuista on osoitteessa <http://firefogg.org/sites>.

KOODAUS VLC:LLÄ

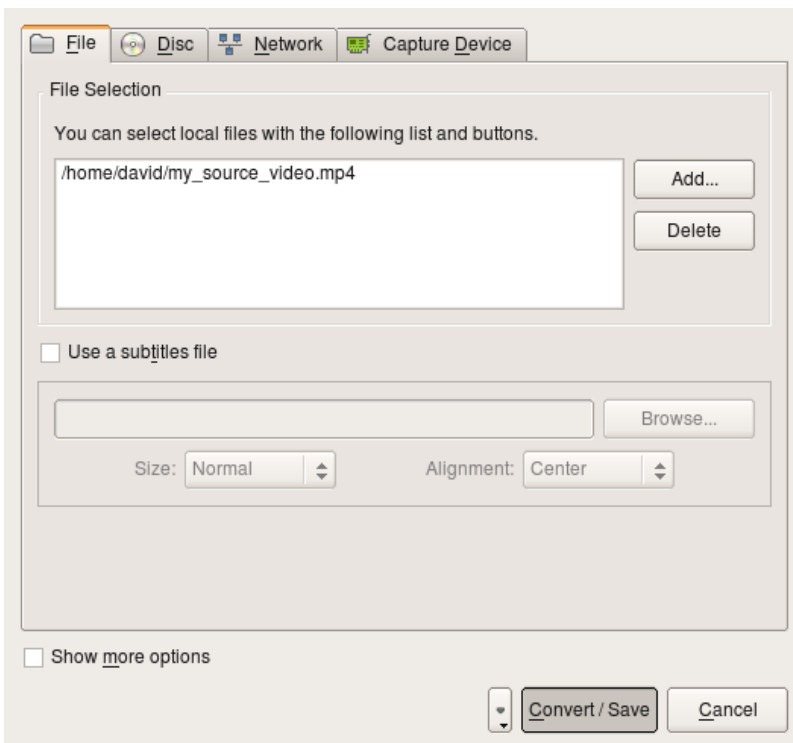
VLC-mediasoitin (www.videolan.org) mahdollistaa videotiedostojen helpon koodauksen Theora-muotoon. Koodaus voidaan tehdä joko graafisen käyttöliittymän kautta tai komentoriviltä. Seuraavat ohjeet on kirjoitettu Ubuntussa toimivaa VLC:n 1.0.1 versiota varten.

GRAAFISEN KÄYTTÖLIITTYMÄN KÄYTTÖ

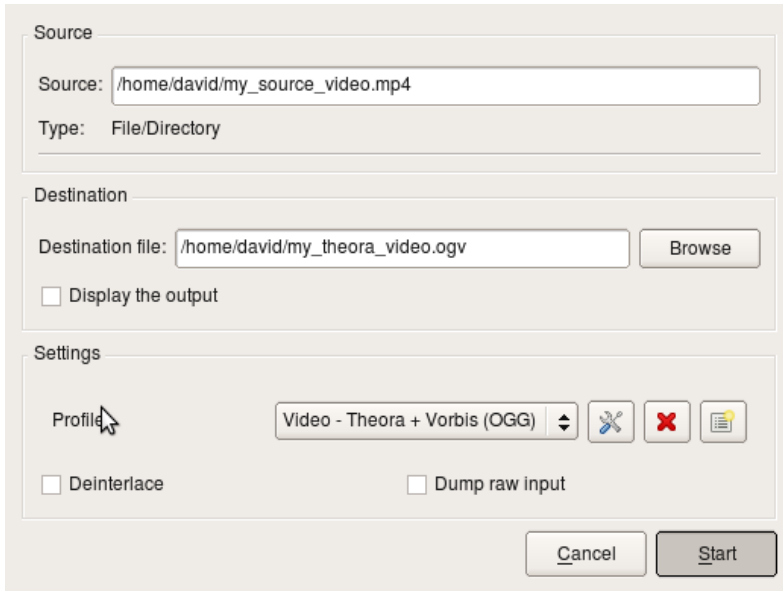
Kun käynnistät VLC-mediasoittimen, sinut viedään suoraan sen pääikkunaan:



Media-valikossa voit valita *Convert/Save*. Tämä avaa seuraavan dialogin:

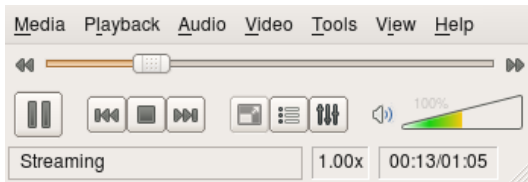


Kohdassa *File Selection* klikkaa *Add* ja valitse lähdevideo koodattavaksi. Sen jälkeen voit klikata *Convert/Save* valikon alaosassa. Tämä vie sinut koodausdialogiin:



Kohdassa *Destination* klikkaa *Browse* ja valitse sijainti sille Theora-tiedostolle, jonka tahdot luoda. Muistakaa, että oikea tiedostonimen laajennus Theora-tiedostolle on **.ogv**. Kohdassa *Settings* aseta profiili kohtaan "Video - Theora + Vorbis (OGG)".

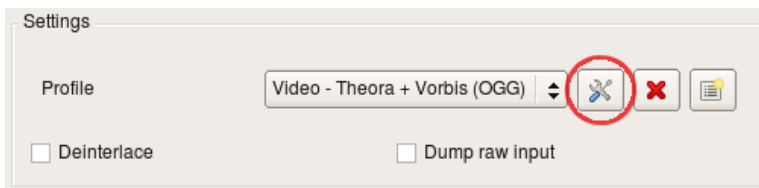
Sen jälkeen paina *Start* -nappia. Tämä aloittaa koodausprosessin ja vie sinut takaisin VLC:n päädialogiin:



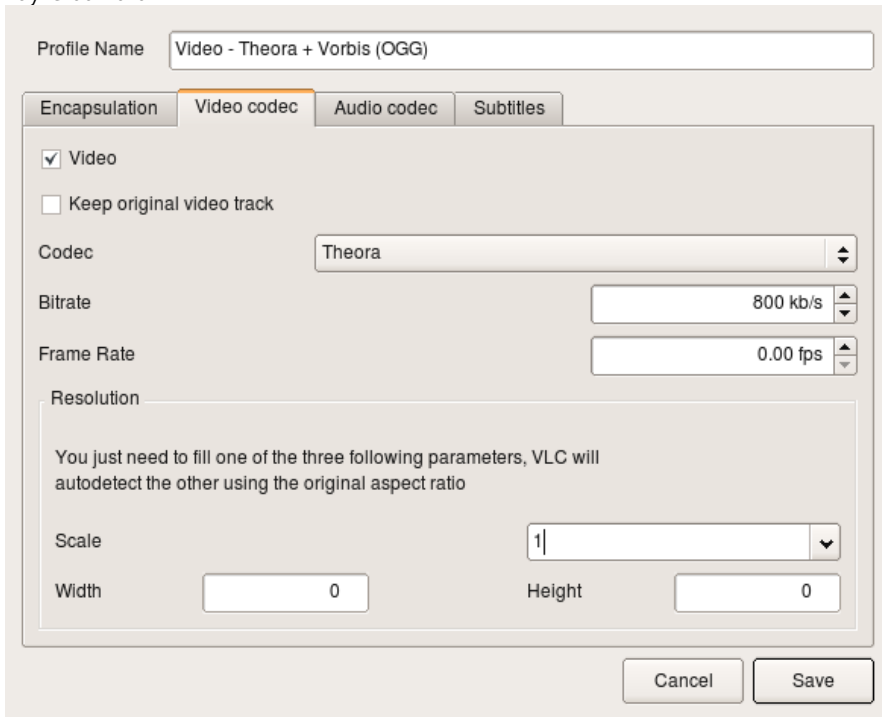
Ikununan alaosassa on näkyvissä teksti "Streaming". Tämä osoittaa, että se on nopeasti koodaamassa tiedostoasi. Liukusäädin liikkuu hitaasti oikealle koodauksen edistyessä. Kun koodaus on valmis, liukusäädin siirtyy takaisin vasemmalle ja "Streaming" -teksti katoaa.

KEHITTYYNEEMMÄT VAIHTOEHDOT

Jos et ole tyytyväinen koodauksen lopputulokseen, yritä muuttaa joitain kehittyneempiä koodausparametrejä. Edellisessä koodausdialogissa, ennen Startin painamista, paina nappia, jossa on työkaluikoni, juuri vasemmalle profiilin valinnasta:



Tämä avaa uuden dialogin, jossa on 4 välilehtä, jotka on nimetty; "Encapsulation", "Video codec", "Audio codec" ja "Subtitles". Varmista, että et muuta mitään parametrejä kohdassa "Encapsulation". Tarvitsemasi videon koodausoptiot ovat "Video codec" -välilehdessä, joka näytetään alla:



Jos koodauksen tuloksena oleva laatu on hyvin heikko, yritä nostaa *Bitratea*. Jos videon lähdemateriaalilla on hyvin korkea resoluutio, yritä asettaa *Scale* arvoon 0.5 koodataksesi puolella alkuperäisestä resoluutiosta. Tätä kirjoitettaessa videon resoluution muuttaminen ei toimi kunnolla kaikelle lähdemateriaalille, kun VLC 1.0.1 on käytössä.

KOODAUS KOMENTORIVILTÄ

Jos koodaat suuren joukon tiedostoja Theora-muotoon, VLC-dialogin läpi klikkailu voi käydä pitkästyttäväksi ja virheitä tulee helposti. VLC:n komentorivikäyttöliittymä pelastaa tältä. Vaikka se ei ole yhtä intuitiivinen kuin graafinen käyttöliittymä, se antaa sinun toistaa täsmälleen koodausprosessin, jossa on pysyvät parametrit.

Seuraavien komentoja voit käyttää koodataksesi lähdevideoasi (tässä tapauksessa tiedostoja kutsutaan nimillä "my_source_video.mp4" ja "my_theora_video.ogv" you can of course, use whatever name you want) samoilla parametreillä, joita käytetään graafisessa käyttöliittymässä yllä:

```
vlc my_source_video.mp4 \
--sout="#transcode{vcodec=theo,vb=800,scale=1,deinterlace=0,\
acodec=vorb,ab=128,channels=2,samplerate=44100}\
:standard{access=file,mux=ogg,dst='my_theora_video.ogv'}"
```

Kaikki aiemmin kehittyneemmissä koodausoptioissa määritellyt parametrit annetaan nyt tekstimuodossa. Ainoat merkitykselliset optiot, joiden merkitys ei ole heti selvä, ovat vb, joka merkitsee videon bittinopeutta, ja ab, joka merkitsee koodatun audion bittinopeutta.

Suosittelut koodausoptiot

Yllä näytetty komentorivi pelkästään käyttää parametrejä, jotka annettiin graafisessa käyttöliittymässä, mikä ei ole mitenkään optimoitua Theoran koodaukseen. Voimme tehdä paljon paremmin käyttämällä optioita, jotka ovat tarjolla pelkästään komentorivillä. Parempi tapa voisi olla käyttää seuraavaa komentoriviä pohjana koodaukselle. Muuttele sitä sopimaan tarpeisiisi:

```
vlc my_source_video.mp4 \
--sout-theora-quality=5 \
--sout-vorbis-quality=1 \
--sout="#transcode{venc=theora,vcodec=theo,scale=0.1,deinterlace=0,\
    croptop=0,cropbottom=0,cropleft=0,cropright=0,\
    acodec=vorb,channels=2,samplerate=44100}\
:standard{access=file,mux=ogg,dst='my_theora_video.ogv'}"
```

Tässä esimerkissä videon ja audion laatu on spesifioitu numeroilla, jotka jatkuvat arvosta 0 (matalin laatu) arvoon 10 (korkein laatu). Jos tahdot poistaa mustat tai häiriöiset reunat videon ympäriltä, käytä optioita välillä `croptop` ja `cropright`.

Nämä esimerkit vaativat, että VLC:n asennuksesi tulee VLC Theora -pluginin kanssa. Varmista pluginin olemassaolo kirjoittamalla:

```
vlc -p theora
```

Vaikka tämä printtaisi *"No matching module found"* voi edelleenkin olla mahdollista koodata se Theoraan, käyttämällä `ffmpeg`-pluginia, joka tukee myös Theoraa. Kuitenkin kehittyneempi `--sout-theora-quality` -optio ei ole tarjolla `ffmpeg`issä.

Mikset käyttäisi graafista käyttöliittymää

Theora-koodekki toimii parhaiten, kun koodataan määritellylle videolaadulle. Koodaus annetulle kohdebittinopeudelle antaa aina huonompia tuloksia samalle tiedostokoolle. Valitettavasti ei ole tapaa määritellä videon kohdelaatua graafisen käyttöliittymän kautta, minkä vuoksi sinun ei pitäisi käyttää sitä mihinkään professionaaliseen koodaustyöhön. Huomaa myös, että et voi leikata videon reunoja kun koodaat graafisesta käyttöliittymästä.

FFMPEG2THEORA

ffmpeg2theora on hyvin kehittynyt Theoran koodaussovellus. Kehittyneempi toiminnallisuus tulee sillä hinnalla, että on pakko käyttää komentoriviä, koska graafista käyttöliittymää ei ole tarjolla Mac OS X ja Windows -käyttöjärjestelmiin voit ladata ffmpeg2theoran osoitteesta <http://v2v.cc/~j/ffmpeg2theora/download.html>. Jos käytät GNU/Linuxin uutta versiota, on hyvin todennäköistä, että distrosi tulee ffmpeg2theora-pakettien kanssa, tai että ne voidaan asentaa distron ohjelmistopakettien hallintaohjelmalla.

PERUSKÄYTTÖ

Komentorivillä voit kirjoittaa:

```
ffmpeg2theora my_source_video.mp4 -o my_theora_video.ogv
```

Tämä koodaa lähdetiedoston "my_source_video.mp4" ja luo uuden Theora-videotiedoston "my_theora_video.ogv".

PARAMETRIEN LISÄÄMINEN

Jos et ole tyytyväinen koodauksen tuloksiin, on aika alkaa säätämään koodaamisen parametrejä. Aloitamme säätämällä koodatun videon laatua. Laatu annetaan numerolla: 0 on alhaisin laatu ja pienin tiedosto, 10 on korkein laatu ja suurin tiedosto. Yritä koodata tämä korkealaatuisena videona, jonka laatu on 9, ja korkealaatuisena äänen laatuna 6:

```
ffmpeg2theora my_source_video.mp4 -o my_theora_video.ogv \
--videoquality 9 --audioquality 6
```

Seuraavat esimerkit näyttävät koodauksen perusparametrit. Kopioi ja liimaa se ja muokkaa sitä sopimaan tarpeisiisi:

```
ffmpeg2theora my_source_video.mp4 -o my_theora_video.ogv \
--videoquality 9 --audioquality 6 \
--croptop 0 --cropbottom 0 --cropleft 0 --cropright 0 \
--width 720 --height 576 \
--title "Video Title" --artist "Artist Name" --date "1997-12-31"
```

Jos et tahdo skaalata videoruudun kokoa, pudota optiot `--width` ja `--height`. Ei ole tapaa määritellä skaalaa, joten katso sisään tulevan videon kokoa ja laske kohteen ruutukoko tarpeen mukaan. Useimmissa tapauksissa on paras vain määritellä yksi optioista `--width` tai `--height`, puuttuva optio säädetään sitten automaattisesti oikeaan arvoon.

KEHITTYNEEMMÄT OPTIOT

Ffmpeg2theora tukee monia muita parametrejä kehittyneempään käyttöön, niitä kaikkia ei voida kuvailla tässä yksityiskohtaisesti. Saadaksesi yleiskuvan kaikista tarjolla olevista optioista, kirjoita:

```
ffmpeg2theora --help
```

Riippuen käyttämästäsi käyttöjärjestelmästä, voit kyetä avaamaan ffmpeg2theora -manuaalin kirjoittamalla

```
man ffmpeg2theora
```

Seuraavat optiot ovat usein käytökelpoisia:

```
--sync
```

Kopioi mikä tahansa lähdevideon audio-video synkronisaatio kohteena olevaan Theora-videoon. Riippuen käytetystä lähdevideosta tämä voi korjata audio-videon viivästyksen ongelmat, jotka johtuvat koodausprosessista.

--keyint

Aseta avainruutujen väli, esimerkiksi ruutujen määrä avainruutujen välissä. Suuren -kokoiset arvot johtavat pienempään videon kokoon, kuitenkin etsiminen ja leikkaaminen ei toimi hyvin Theora-tiedostoissa, joissa on suuri avainruutujen väli.

--framerate

Asettaa kuvataajuuden luodulle videotiedostolle. Jos yrität luoda Theora-tiedostoja, joiden tiedostokoko on hyvin pieni, yritä tarkentaa puolet videon kuvantaajuudesta.

--starttime --endtime

Nämä kaksi vaihtoehtoa antavat sinun kopioida vain osan lähdevideosta kun koodaat. Määritä ja määräksi sekunteja videon alusta.

2-PASS KOODAUS

Tuleva versio 0.25 ffmpeg2theorasta tulee tukemaan 2-pass koodaustilaa, jota kuvaillaan tässä osassa. Kun versio 0.25 julkaistaan, lataa se osoitteesta <http://v2v.cc/~j/ffmpeg2theora/download.html>. Nämä esimerkit eivät toimi vanhemmilla versioilla.

Miksi 2-Pass -koodaus

2-Pass -koodaus on kohuttu. Monet ihmiset ajattelevat, että joudut koodaamaan kaksi kertaa saadaksesi jatkuvan subjektiivisen laadun kokemuksen koko videon ajan. Näin sitä käytetään monissa epävapaisissa videokoodekeissa, kuten DivX:ssä. Kuitenkin, kuten olemme nähneet, ffmpeg2theora kykenee hyvin koodaamaan jatkuvaa kohdelaatua varten yhdellä kerralla käyttäen optiota *--videoquality*.

Ainoa todellinen etu 2-Pass tilan käytössä sen sijaan että käyttäisi optiota *--videoquality*, on kyky luoda annetun kokoinen Theora-videotiedosto. Ajattele, että tahdot koodata videon, jonka täytyy mahtua yhdelle CD-levylle, jossa on 700 MB tilaa tarjolla. Jos tahdot jatkuvana pysyvän videon laadun, mutta etukäteen et voi arvata mikä *--videoquality* osuu täsmälleen kokoon 700 MB. 2-Pass -tilan käyttö sopii tähän täsmälleen.

2-Pass tilan käyttö

Jos tahdot koodata videon "my_source_video.mp4" Theora-videoksi, niin että tiedostokoko on täsmälleen 700MB. ffmpeg2theora ei salli koodatun tiedoston suoraa määrittelyä. Sen sijaan tarkennat keskimääräisen videon bittinopeuden videota varten. Myös audio vaatii dataa, joka täytyy ottaa huomioon.

Päätetäänksemme videon keskimääräisen bittinopeuden joudumme ensin etsimään lähdevideon "my_source_video.mp4" koon. Ffmpeg2theora voi auttaa meitä tässä. Kirjoita:

```
ffmpeg2theora --info "my_source_video.mp4"
```

Mikä printtaa muun tiedon mukana:

```
{
  "duration": 2365.165100,
  "bitrate": 6437.331055,
  [..]
}
```

Emme voi käyttää koko 2368 kBit/s pelkälle videollemme. Meillä on mhyös ääniraita, joka tarvitsee 128kBit/s. Kun vähennämme tästä 1% ottaaksemme huomioon äänen ja videon kapsuloinnissa jääneen tilan, meille jää 2218 kBit/s videolle ja 128 kBit/s äänelle.

Seuraava komento tekee 2-pass -enkoodauksen, luoden Theoran videotiedoston "my_theora_video.ogv":

```
ffmpeg2theora my_source_video.mp4 -o my_theora_video.ogv \  
--two-pass --videobitrate 2218 --audiobitrate 128
```

Huomaa, että toisin kuin muissa 2-pas -enkoodaajissa, ffmpeg2theora -komento täytyy antaa vain kerran. Jos sinun täytyy hallita prosessia tarkemmin ja tehdä kroppausta, skaalausta jne., voit vapaasti kopioida muita vaihtoehtoja edellisissä osissa annetuista esimerkeistä.

THOGGEN

Thoggen (<http://www.thoggen.net/>) on yksinkertainen, helposti käytettävä DVD-imuohjelma, jolla voi luoda Theora-videoita DVD-levyistä. Huomaa, että ainoastaan video-osa DVD:stä voidaan koodata, mukana olevat valikot jäävät pois.

DVD:n koodaus Thoggenia käyttäen sisältää kaksi askelta:

1. Valitaan DVD-otsikot imettäväksi ja koodattaviksi
2. Konfiguroidaan koodausprosessin parametrit

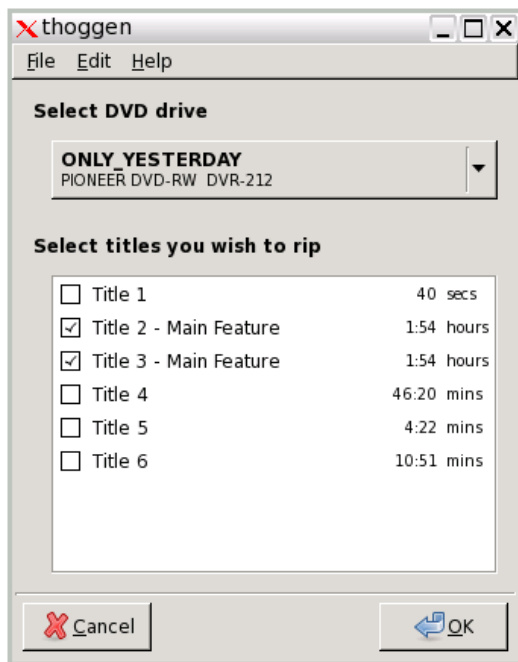
Jos Thoggenia ei ole asennettu, sinun täytyy toki tehdä se ensin. Jos käytät Ubuntuä, kirjoita tämä terminäali-ikkunaan:

```
sudo apt-get install thoggen
```

Salasanaasi kysytään, tämän jälkeen Thoggen asennetaan.

DVD-OTSIKOIDEN VALINTA KOODATTAVIKSI

Aloituksen jälkeen Thoggen löytää automaattisesti DVD-median, joka on missä tahansa DVD-soittimessa, ja esittää listan levyllä olevista otsikoista:



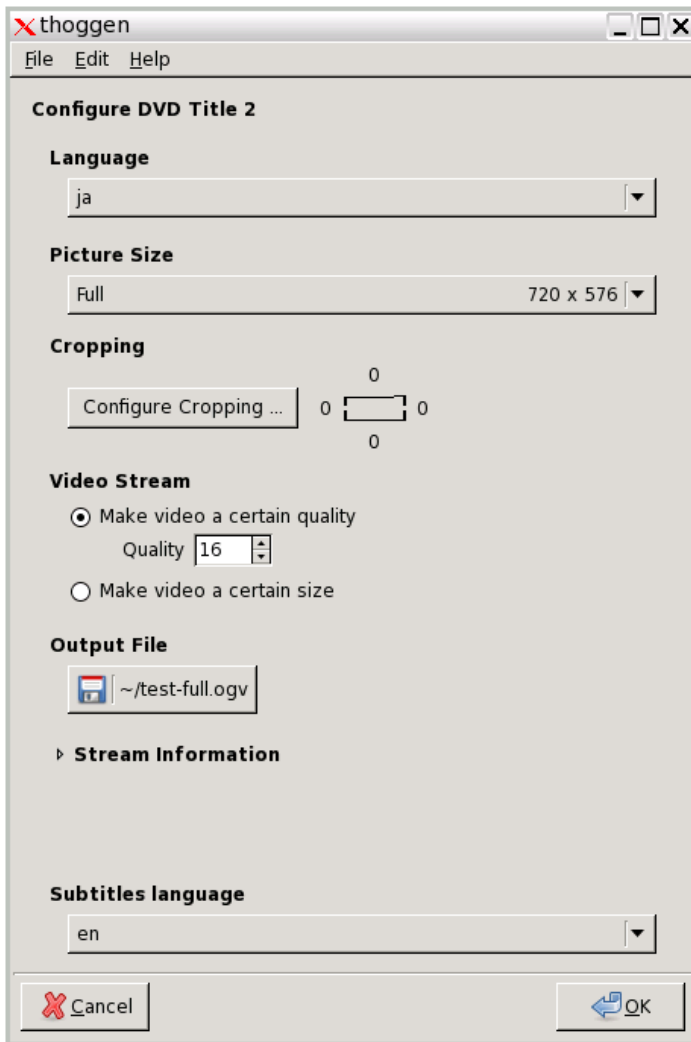
Sinua pyydetään valitsemaan otsikot, jotka imetään ja muutetaan Theoraan. Pisin otsikko on valittu automaattisesti, ja tämä on yleensä päävideo. Jos et tahdo tätä tiedostoa, mutta se sopimaan tarpeisiisi.

Huomaa, että voit myös konvertoida DVD-kuvan kovalevylläsi, sen sijaan että konvertoisit sen suoraan DVD:ltä. Tehdäksesi tämän valitse kuvan sijainti "Tiedosto"-valikosta. Sama otsikkojen lista näytetään, aivan kuin se olisi otettu fyysiseltä DVD:ltä.

Kun tämä on tehty, paina OK ja sinut viedään seuraavaan dialogiin.

KOODAUSPROSESSIN ASETUKSET

Seuraavan dialogin avulla voit määrittellä koodausprosessin parametrejä:



Jos olet valinnut enemmän kuin yhden videon edellisessä askeleessa, tämä ikkuna näytetään kerran jokaiselle niistä. Monikieliselle DVD:lle voit valita, minkä kielen sisällytät. Valitse laatu ja videon koko, sekä minne tallennat videon. Jos et ole varma, minkä option tahdot valita, käytä vain ennakko-oletuksia. Huomaa, että videon laatu on esitetty numeroina 1 (heikoin laatu, pienin tiedosto) - 62 (korkein laatu, suurin tiedosto).

Kun olet tyytyväinen asetuksiin, paina OK aloittaaksesi koodauksen. Silloin sinulle näytetään diaesitys videosta tilanteen näyttävän edistyspalkin kanssa, samalla kun odotat koodausprosessin päättymistä.

Nyt voit käyttää VLC:tä, tai mitä tahansa muuta Theoraa tukevaa soitinta, nähdäksesi kopiosi DVD:stä, ilman että sinun täytyy itse asiassa etsiä ja asentaa DVD asemaan, kun tahdot katsoa elokuvasi.

HUOMIO RAJAUKSESTA

Thoggen voi poistaa reunat vasemmalla, oikealla, ylhäällä ja alhaalla koodatessaan DVD-tiedostoa, tämä mahdollisuus löytyy Configure Cropping -nappulasta. On usein hyvä idea poistaa noin 5% elokuvan leveydestä ja korkeudesta. Tämä johtuu siitä, että DVD:t on valmistettu ikivanhoja analogitelevisioita varten, joten elokuvien ympärillä on usein rajat. Vanhemmat DVD:t sisältävät usein häiriöitä ja hälyä elokuvan reunoissa, ja tämä näkyy vain katsottaessa niitä DVD:ltä PC:llä.

Tyypillistä DVD-resoluutiota 720x576 pikseliä varten kannattaa kokeilla 8 pikselin poistamista vasemmalta, oikealta, alhaalta ja ylhäältä, jolloin ruudun kooksi jää 704x560 pikseliä.

TEKSTITYKSET

19. JOHDANTO

20. JAKELU

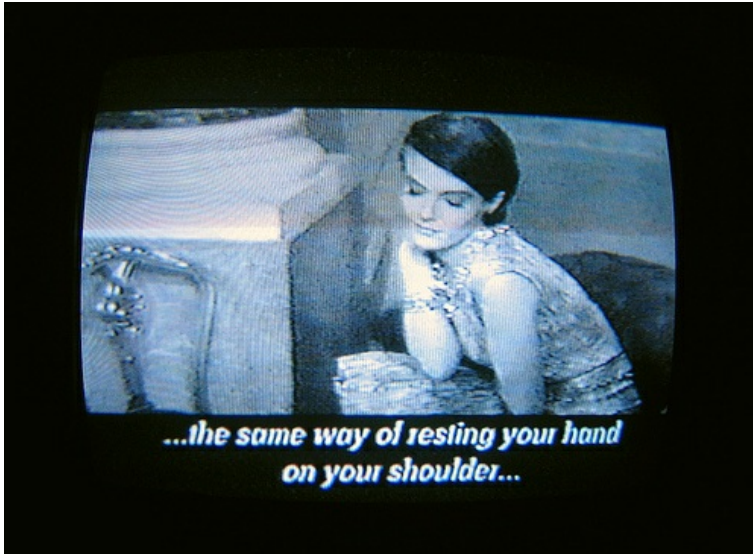
21. Tekstityksien lisääminen

22. TEKSTITYKSIEN TOISTAMINEN

23. JULKAISEMINEN

TEKSTITYKSET

Useimmissa videoissa on ihmisiä, jotka puhuvat tietyllä kielellä; jotta näistä videoista saataisiin käytettäviä ja ymmärrettäviä globaalille yleisölle, video täytyy tekstittää tai dubata. Tekstitykset on paljon helpompi tehdä: äänen dubbaus vaatii aikaa ja kokemusta ohjelmista, mutta tekstitykset voit tehdä pelkällä videosoittimella ja tekstieditorilla.



TEKSTITYKSIEN ETSIMINEN

Ennen tekstitysprojektin aloittamista kannattaa etsiä netistä olemassa olevia tekstitystiedostoja, varsinkin jos video on tunnettu tai kaupallinen työ. Esimerkiksi, jos liität kohtauksen amerikkalaisesta dokumenttielokuvasta videoon, on olemassa resursseja, joista voit etsiä tekstityksiä tälle materiaalille. Kuitenkin on vaikea löytää valmiita tekstityksiä muille kuin tunnetuille videoille tai elokuville, ja avoimen lisenssin tekstityksiä on vielä vaikeampi löytää.

On muutamia seikkoja, joihin törmää helposti etsiessään tekstityksiä. Elokuvista on melkein aina monta erilaista versiota. Voidaan kuvitella, että mikä tahansa ylimääräinen kohtaus, pidennetyt alkutekstit, tai formaatin muutos voi muuttaa tekstityksien ajastusta, mikä yleensä tekee tekstityksistä hyödyttömiä. On siis tärkeää etsiä tietyn elokuvaversioon ääneen sopivia tekstityksiä. On olemassa vapaita ohjelmistotyökaluja, kuten Sub Downloader (<http://www.subdownloader.net/>), joka helpottaa ongelmaa sovittamalla tekstitykset tiettyihin elokuvaversioihin. Toinen ongelma on tekstitystiedoston formaatti. On olemassa erilaisia formaatteja erilaisille videotyypeille ja erilaisille fyysisen median tyypeille (HD, DVD, Blu Ray jne.), mikä vaikuttaa tekstityksien valintaan tietylle filmille.

Tässä on resursseja tekstityksien löytämiseen:

- OpenSubtitles.org: <http://www.opensubtitles.org/en>
- TinyTM: <http://tinytm.sourceforge.net/>
- DivX Subtitles: <http://www.divxsubtitles.net/>
- AllSubs.org <http://www.allsubs.org>

TEKSTITYSTIEDOSTOJEN FORMAATIT

Tekstitystiedoston formaatti tarkoittaa tiedoston formaatin (teksti tai kuva), joka sisältää tekstitykset ja ajastustiedot. Jotkin tekstiin pohjautuvat formaatit mahdollistavat tyylietietojen tarkentamisen, kuten värien tai tekstityksen sijainnin.

Tässä on joitain tekstitystiedostojen formaatteja:

1. Micro DVD (.sub) - tekstiin perustuva formaatti, jossa on videoruutuun perustuva ajoitus, ja ei tekstin tyyliä.
2. Sub Rip (.srt) - tekstiin perustuva formaatti, jossa on videon keston perustuva ajoitus, eikä tekstin tyyliä.
3. VOB Sub (.sub, .idx) - kuvaan perustuva formaatti, käytetään yleensä DVD:ssä.
4. Sub Station Alpha / Advanced Sub Station (.ssa, .ass) - tekstiin perustuva formaatti, jossa on videon pituuden mukainen ajoitus, ja tekstin tyyli- ja metadatatietoattribuutit.
5. Sub Viewer (.sub) - tekstiin perustuva formaatti, jossa on videon pituuteen perustuva ajoitus, tekstin tyyli- ja metadatatietoattribuutit.

Me keskitymme vain tekstityformaattiin Sub Rip (.srt), jota tukevat useimmat videosoittimet ja tekstityseitorit. SRT -tiedostoja voi myös editoida tekstieditoreilla, tai kehittyneemmillä ohjelmilla, kuten Jubler, GnomeSubtitle, Gaupol tai SubtitleEditor.

SRT-TIEDOSTOJEN EDITOINTI

SRT-tekstitystiedosto on vain tekstitiedosto, joka on formatoitu yksinkertaisella tavalla, jotta soitin voi lukea sen ja esittää tekstitykset silloin, kun ne pitäisi toistaa videossa. SRT on hyvin yksinkertainen, laajalti käytetty tekstityformaatti. Jos löydät olemassaolevan SRT-tiedoston videolle, joka sinun täytyy tekstittää, on helppoa luoda tekstitykset muille kielille, kun tiedät kuinka SRT-tiedosto toimii.

SRT-tiedosto on lista rivejä, jotka näyttävät tältä:

```
1
00:03:05,260 --> 00:03:07,920
Hei maailma.
```

Ensimmäinen rivi on tekstityksen numero, joka alkaa numerosta 1 ja jatkuu niin pitkälle kuin tarvitaan. Seuraava rivi on aika, jolloin tekstitys ilmestyy ja katoaa videossa, ja se on merkitty muotoon tunnit:minuutit:sekuntit,millisekuntit. Kolmas rivi ja sen jälkeen tulevat rivit ensimmäiseen tyhjään riviin asti ovat varsinainen tekstitys. Yksi tyhjä rivi tarvitaan merkitsemään tekstin loppumista. Voit lisätä kuinka monta kolmoisriviä tahansa, niin monta kuin tarvitset videoosi.

Edellisessä esimerkissä se tarkoittaa vain, että ensimmäinen tekstitys ilmestyy ajankohtana 3 minuuttia ja 5,26 sekuntia, katoaa kohdassa 3 minuuttia ja 7,92 sekuntia, ja tekstinä on "Hei maailma". Siinäpä se.

UUDEN TEKSTITYKSEN LUOMINEN

Luodaksesi uuden tekstitystiedoston tahdot ehkä kehittyneemmän työkalun, jolla on helppo liittää tekstitykset tiettyihin kohtiin videossa. Jubler, GnomeSubtitle, Gaupol ja SubtitleEditor ovat kaikki vapaita ohjelmistotyökaluja ja tsekkaamisen arvoisia.

JAKELU

Tekstitysformaattien, kuten .SRT, käyttö merkitsee, että voit jaella tekstityksiä monille eri kielille jakamatta eri versiota videostasi joka kielell. Tarvitset vain erilliset .SRT -tekstitystiedostot jokaiselle vaaditulle kielelle, ja voit laittaa nuo tiedostot verkkoon.

Tämä strategia on hyvin yleinen tekstitysten maailmassa. Tekstitykset ovat mukana erillisinä tiedostoina, mikä mahdollistaa tiedoston käytön, muuttamisen tai poistamisen ilman että videotiedostoon itseensä vaikutettaisiin mitenkään. Tekniikan heikkoutena on, että tekstitysformaattista itsestään tulee ongelma. Soittimien täytyy hyväksyä formaatti näyttääkseen tekstitykset oikein. Ja käyttäjien täytyy tietää hieman tekstitysten toiminnasta, jotta ne voivat soittaa tekstitystiedostot oikein. Jos jakelet .SRT -tiedostoja ladattavalla videolla, tarkasta, että mukana on ohjeet tekstitysten lataamiseen ja toistamiseen. Jos jakelet videota verkossa, voit käyttää HTML5 ja JavaScript -koodia jaellaksesi eri tekstitystiedostoja verkkosivullasi.

On myös mahdollista tutkia monien .SRT-tiedostojen laittamista videotiedoston sisään. Tämä antaa käyttäjälle mahdollisuuden valita käännöksistä, jotka olet laittanut tarjolle (tai olla näyttämättä mitään tekstityksiä) ilman tarvetta muihin tekstitystiedostoihin. Tätä tukeviin patenttivapaisiin videosäilöformaatteihin kuuluvat Matroska Multimedia Container (MKV) ja Ogg -säilöformaatit.

21. TEKSTITYKSIEN LISÄÄMINEN

Jos tahdot videotiedostosi sisältävän tekstitystiedoston, jotta et joudu jakelemaan .srt-tiedostoa erikseen, joudut liittämään tekstitystiedoston videoon. Videon koodaustyökalu **ffmpeg2theora** sisältää muutaman komentorivioption, joilla voit liittää tekstitykset videoosi.

ffmpeg2theora on tarjolla useimpiin käyttöjärjestelmiin, kuten Mac OS X ja GNU/Linux.

Tekstityksiin liittyy kolme tärkeää komentoa:

- `--subtitles` osoittaa SRT-muotoiseen tekstitystiedostoon,
- `--subtitles-language` määrittää tekstitysten kielen,
- `--subtitles-encoding` määrittelee tekstitystiedoston kirjoitusmerkit.

Katsotaanpa `ffmpeg2theora` -komennon optioiden käyttöä `srt`-tiedostojen liittämiseksi Theora-tiedostoon.

subtitles-language - Tämä optio asettaa määritellyn kielen. Jokaisella kielellä on standardikoodi, joka auttaa ihmisiä kuvailemaan kielen, mikä tahansa heidän kieltänsä onkin. Esimerkiksi englanniksi saksan kieli on `german`, mutta suomeksi se on `saksa`. Hämmennyksen välttämiseksi kansainvälinen ISO 639-1 -standardi määrittelee jokaisen kielen kahden kirjaimen koodilla. Saksan koodi on `"de"`.

subtitles-encoding - Tämä optio määrittelee tekstin koodauksen standardin, tämä on välttämätöntä erilaisten kirjoitusjärjestelmien vuoksi. Tietokoneet käyttivät pitkään 7-bittisiä merkkistöjä, joissa oli vain 127 kirjoitusmerkkiä, edustamaan kirjaimia ja muita symboleja. Esimerkiksi US-ASCII -merkkistössä on 94 tulostuvaa merkkiä ja 33 kontrollikoodia. Lukemattomia 8-bittisiä merkkistöjä, joissa on 256 kirjoitusmerkkiä, on ilmestynyt tämän jälkeen edustamaan eri aakkosia ja syllabisia kirjoitusjärjestelmiä. Monet 16-bittiset kirjoitusjärjestelmät mahdollistavat kiinalaisiin merkkeihin perustuvat kirjoitusjärjestelmät. Kuitenkaan 7 tai 8 bittiä ei ole tarpeeksi kaikkia yhden kirjoitusjärjestelmän merkkejä varten, eikä varsinkaan kaikille nykyisin käytössä oleville kirjoitusjärjestelmille. Ihmiset loivat Unicode -merkkistön tukemaan kaikkia kieliä samaan aikaan. Unicoden UTF-8 koodaus on suunniteltu käytettäväksi verkossa, eli järjestelmien välisessä ulkoisessa viestinnässä.

Kuitenkin suuri osa ihmisistä käyttää edelleenkin vanhoja merkkistökoodauksia. Näiden ongelmana on se, että ne voivat mennä päällekkäin, ne käyttävät samoja koodeja täysin erilaisille merkeille. Yleensä väärän koodauksen renderöinnin vuoksi tuloksena on sotkua. Oletusarvoisesti tekstien oletetaan olevan Unicoden UTF-8 koodauksella. Jos ne eivät ole, se täytyy kertoa `ffmpeg2theoralle`. Jos kirjoitat englantia, kirjoitat luultavasti ASCII-merkkistöllä, ISO-8859-1-merkkistöä (Latin-1), tai mahdollisesti Windowsin koodisivua 122. US-ASCII on suunniteltu UTF-8:n alaryhmäksi, joten se toimii, mutta ongelmat alkavat, jos ASCII-laajennukai käytetään Unicode-ympäristössä.

TEKSTITYSTEN LIITTÄMISEN ESIMERKKIKOMENTOJA

Tässä on muutamia esimerkkejä, joissa otetaan `mp4`-videotiedosto (`input.mp4`) ja ulos tulee `ogg`-videotiedosto (`output.ogg`), johon on liitetty tekstitykset:

Jos sinulla on tekstitystiedosto englanniksi (englannin kielikoodi on `"en"`):

```
ffmpeg2theora input.mp4 --subtitles english-subtitles.srt --subtitles-language en -o output.ogg
```

Jos sinulla on espanjankielinen tekstitystiedosto, joka on koodattu `latin1`-merkkistön mukaan:

```
ffmpeg2theora input.mp4 --subtitles spanish.srt --subtitles-language es --subtitles-encoding latin1 -o output.ogv
```

On olemassa muita tekstitysvaihtoehtoja ohjelmalle ffmpeg2theora, mutta nämä ovat tärkeimmät.

TEKSTITYKSIEN LISÄÄMINEN VIDEOON

Jos sinulla on Theora-video ilman sisällytettyjä tekstityksiä, on helppoa myös lisätä niitä, ilman tarvetta koodata video uudelleen. Koska jokainen tekstitysten kieli on tallennettu Ogg-tiedostoon erikseen, niitä voidaan manipuloida erikseen.

Ogg-tiedostoon sisäisesti liitetty tekstitykset koodataan Kate-striiimeiksi. Sellaiset striiimit luodaan ffmpeg2theora-ohjelmalla, mutta ne voidaan myös luoda "raakana" SRT-tiedostosta. Työkalu kateenc tekee tämän. Ubuntussa kateenc on osa kate-tools -pakkausta. Asenna se tällä komennolla:

```
sudo apt-get install libkate-tools
```

Seuraava komento luo englanninkieliset tekstitykset sisältävän striiimin SRT-tiedostosta. Muista, englannin koodi on "en":

```
kateenc -t srt -o english-subtitles.ogg english.srt -c SUB -l en
```

Nyt sinulla on yksi tekstitysstriiimi, jonka voit lisätä Theora-videoosi:

```
oggz-merge -o video-with-subtitles.ogv original-video.ogv english-subtitles.og
```

Ubuntussa oggz-merge on osa oggz-työkalupakettia, jonka asentamiseksi voit tehdä tämän:

```
sudo apt-get install oggz-tools
```

Itse asiassa oggz-työkalut sallivat kaikkien videon eri raitojen voimakkaan manipulaation, joten voit lisätä enemmän kieliä audioraidalle ja niin edelleen.

TEKSTITYKSIEN TOISTAMINEN

On monta helppoa tapaa toistaa tekstityksiä Ogg Theora videossa.

VLC

Asenna ensin VLC verkkosivulta (<http://videolan.org/vlc>) jos et ole jo tehnyt sitä. Nämä ohjeet olettavat, että sinulla on tiedosto tai DVD, jossa on tekstitykset, jotka tahdot näyttää soittaessasi videota.

On kolme tapaa, joilla voit käyttää VLC:tä toistamaan tekstityksiä.

1. DVD:ltä
2. Monikielisestä tiedostosta (kuten Matroska)
3. Erilliseltä tekstitystiedostolta, joka jaetaan videotiedoston kanssa.

TOISTA TEKSTITYKSET DVD-LEVYLTÄ

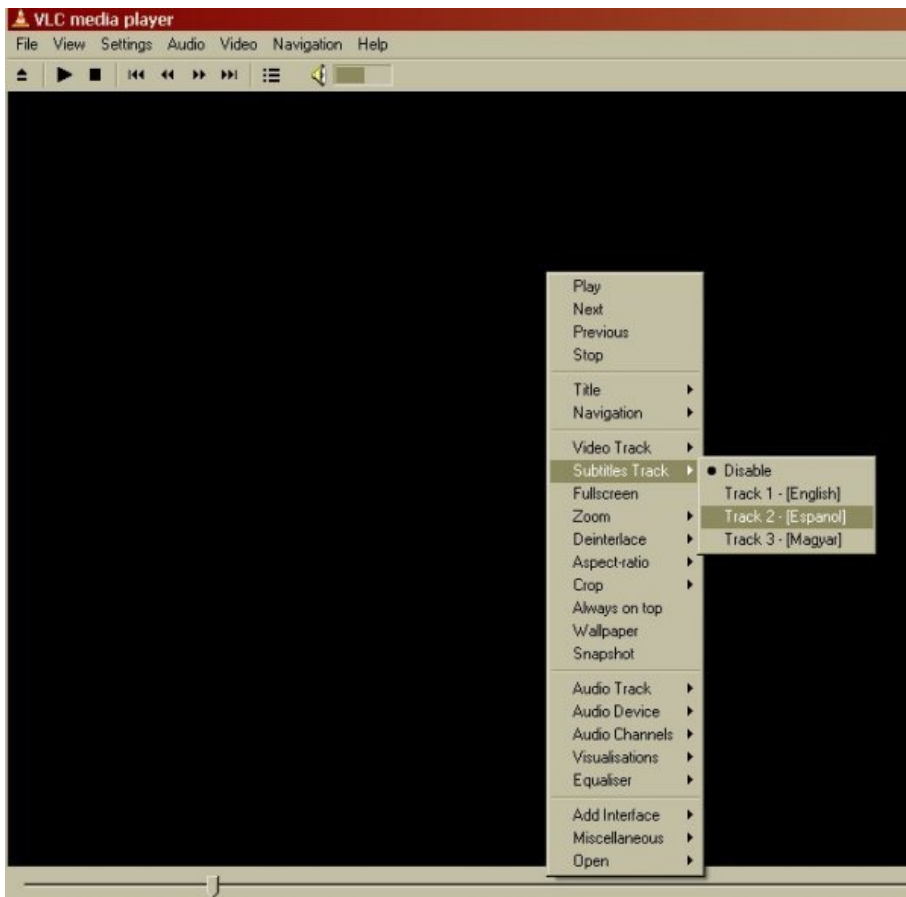
Tehdäksesi tämän laita DVD-levy DVD-soittimeesi. Avaa VLC-soitin ja valitse **File > Open Disk**.



Kirjoita DVD-soittimen kirjain. Se voi ilmestyä automaattisesti. Windowsissa tämä voi olla soitin D:, ja GNU/Linuxissa /media/dvd.



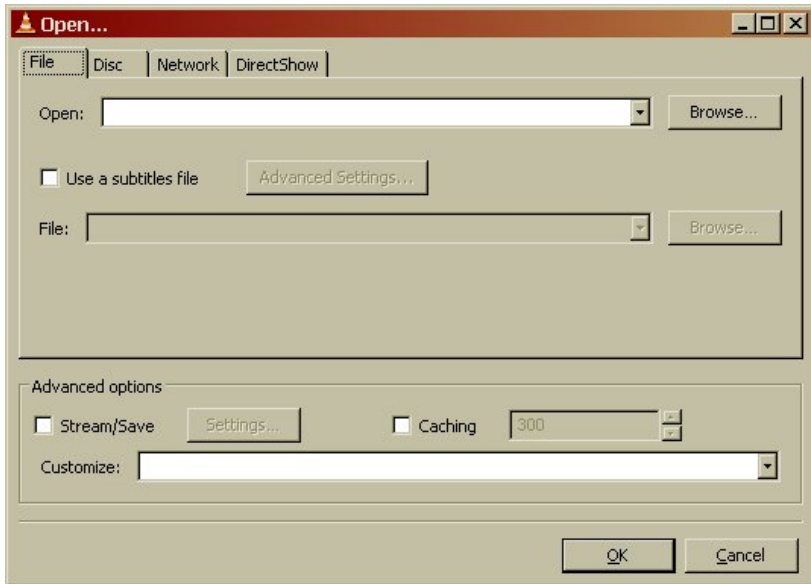
Klikkaa tämän jälkeen OK. DVD:n valikkosivun pitäisi ilmestyä. Klikkaa sitten videota, jota tahdot katsoa. Tämän jälkeen video alkaa nopeasti ja klikkaa oikealla nappulalla videokuvaa. Valitse tekstitysraila, jonka tahdot katsoa.



Tekstityksien pitäisi ilmestyä ruudulle.

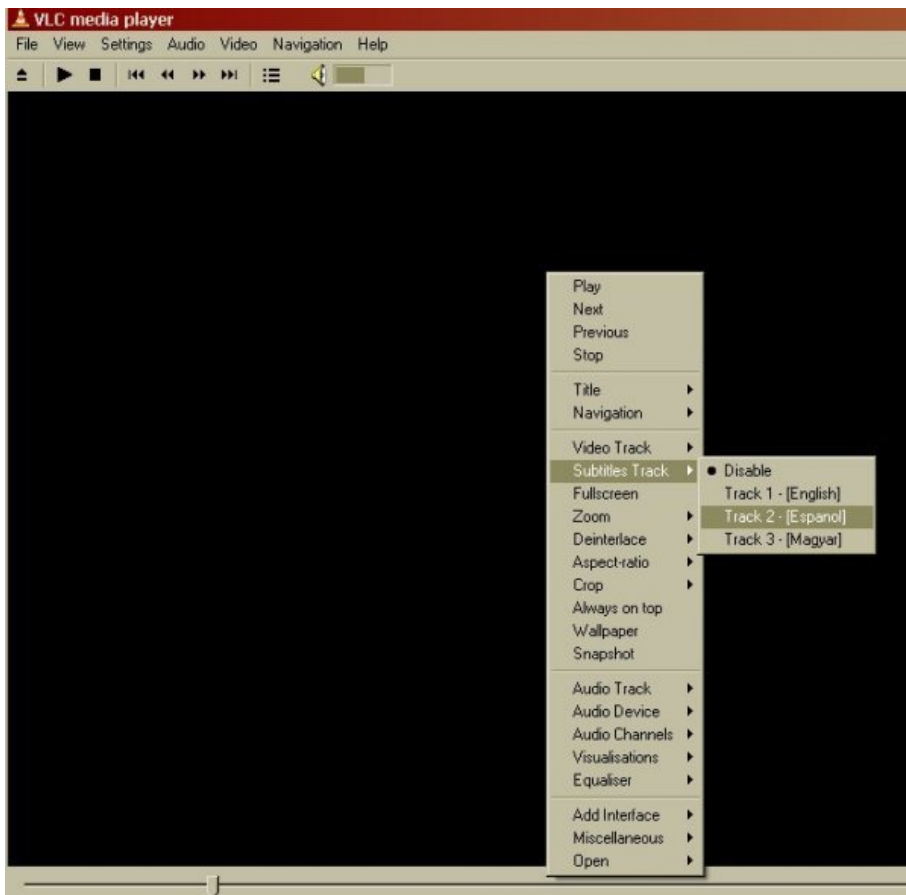
TOISTA TEKSTITYKSET MATROSKA-TIEDOSTOISTA

Tämän tekemisen prosessi on täsmälleen sama kuin yllä, paitsi että prosessia aloitettaessa valitset **File > Open File**. Silloin näet tämän ruudun.



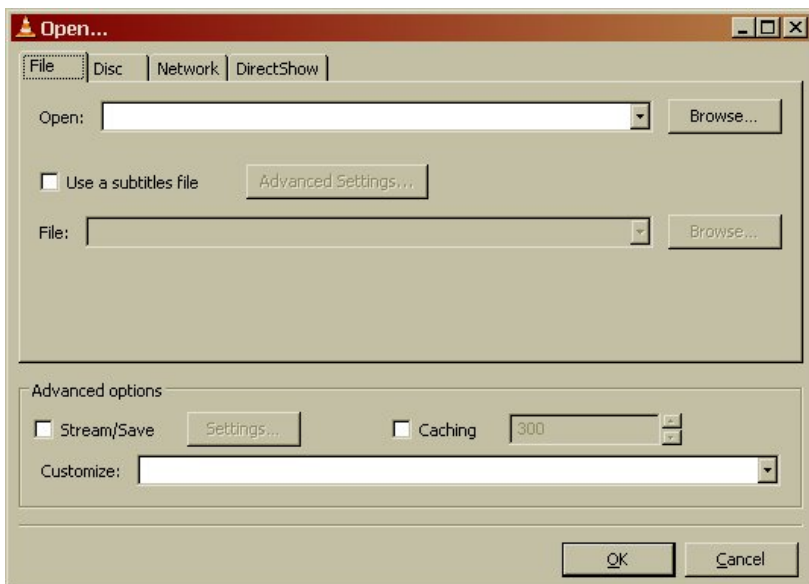
Sinun pitäisi klikata **Browse** -nappia valitaksesi videotiedoston, jonka haluat toistaa. Jos tämä on Matroska-tiedosto, jonka pääte on *.mkv, voit klikata OK selattuasi tätä tiedostoa, sillä siinä on jo tekstitystiedot.

Tämän jälkeen valitse tekstityksen kieli klikkaamalla videoruutua ja valitse **Subtitle Track >** ja valitse kieli.



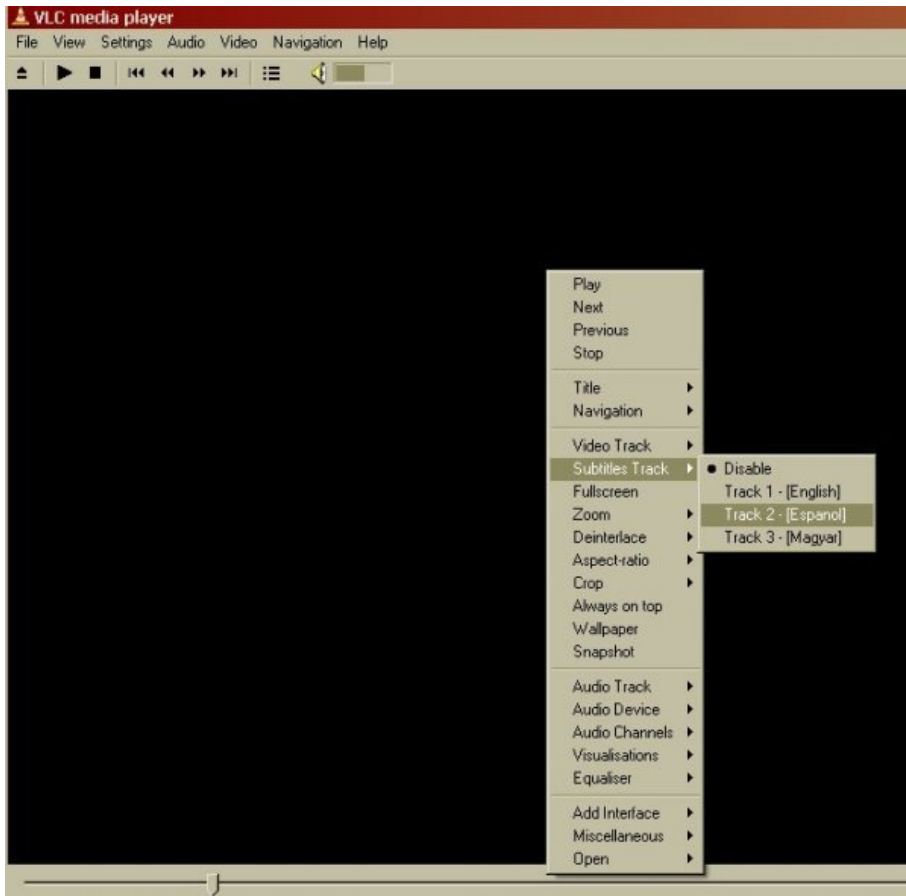
TOISTA ULKOISET TEKSTITYKSET

Jos tahdot toistaa ulkoisen tekstitystiedoston esimerkiksi SRT-tiedostona, valitse **File > Open File**.



Open-laatikossa klikkaa **Browse**-nappia ja valitse videotiedostosi.

Laita rasti laatikkoon **Use a subtitle file** ja klikkaand Browse löytääksesi ulkoisen tekstitystiedoston.



Valitse sitten tekstityksen kieli klikkaamalla oikealla hiiren napilla videoruutua ja valitsemalla **Subtitle Track >** ja valitse sitten tekstitysraila (ulkoiselle tiedostolle, kuten SRT-tiedostolle, on normaalisti vain yksi raita).

JULKAISEMINEN

Riippuen tavasta, jolla loit tekstitykset - liitit ne videoon tai teit erillisen .srt-tiedoston - sinulla on erilaisia tapoja julkaista ne verkossa videosi kanssa.

ULKOISTEN TEKSTITYSTIEDOSTOJEN PITÄMINEN VIDEON KANSSA VERKKOPALVELIMELLA

Oletamme, että sinulla on yksi tai useampia tekstitystiedostoja (SRT-formaatissa) ja video.

HTML5 -videotägi ja Javascript

Voit tarjota Theora -videon esikatselun .srt-tiedoston kanssa ohjelman jquery.srt avulla.

Ensin, huomaa että alla liitämme JQueryn, suosittu GPL Javascript -kirjaston (<http://jquery.com/>), ja esimerkin JavaScript-sovelluksesta, joka näyttää SRT-tekstitykset sivulla. Se on saatavilla: <http://v2v.cc/~i/jquery.srt/jquery.srt.js>

Yksinkertainen HTML-dokumentti näytetään alla, siihen kuuluu modifikaatio JavaScript-tiedostojen liittämiseen ja linkki tekstitystiedostoon. Vain yksi tekstitystiedosto voidaan linkittää kerrallaan, paitsi jos kehität JavaScript-koodia pidemmälle.

Se on pieni skripti, joka lataa .srt-tiedoston ja näyttää sen divin sisällä sivullasi, videon alla tai päällä.

```
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript" src="jquery.srt.js"></script>

<video src="http://example.com/video.ogv" id="video" controls>
</video>
<div class="srt"
  data-video="video"
  data-srt="http://example.com/video.srt" />
```

Voit katsoa tätä Mozillan demoa, jos tahdot esimerkin tekstityksistä monilla kielillä: <http://people.mozilla.com/~prouget/demos/srt/index2.xhtml>

Toinen esimerkki käyttää monia tekstitystiedostoja ja tarjoaa käyttöliittymän niiden valitsemiseen: <http://www.annodex.net/~silvia/itext/>.

TEKSTITETYN VIDEON TARJOAMINEN VERKKOPALVELIMELLA

Jos sinulla on Ogg-tiedostoja, joihin on liitetty tekstitykset, ja tahdot näyttää ne selaimessasi, et voi käyttää videotägiä juuri nyt. Kuitenkaan kaikki selaimet eivät tue videotägiä. Cortado, Java-liitännäinen, joka voi näyttää Theora-videoita, tukee myös liitettyjä tekstityksiä, ja se toimii kaikissa selaimissa, joihin on liitetty Java. Voit hankkia Cortadon uusimman version osoitteesta: <http://theora.org/cortado/>

```
<applet code="com.fluendo.player.Cortado.class" archive="cortado.jar"
  width="512" height="288">
  <param name="url" value="video.ogv"/>
  <param name="katLanguage" value="en">
</applet>
```

Käyttääksesi tekstityksiä Cortadossa annat sille urlin videoon, jossa on tekstitykset, ja samalla tavalla annat parametrin tekstitystiedostolle käytettäväksi. Helpoin tapa on valita kieli, jonka tahdot, katLanguage-parametrin avulla:

```
<param name="katLanguage" value="en">
```

On mahdollista myös muuttaa tai poistaa tekstitys JavaScriptin avulla. Jos tahdot vaihtaa

ranskankielisiin tekstityksiin, sinun tulee asettaa kateLanguage -optio skriptistäsi kutsumalla:

```
document.applets[0].setParam("kateLanguage", "fr");
```

SUORATOISTO

24. MITÄ ON SUORATOISTO

25. ICECAST

26. TSS

27. FFMPEG2THEORA

28. VLC

29. VIRTASTOISTON TALLENTAMINEN

MITÄ SUORATOISTO ON?

Suoratoistoa käytetään kuvailemaan äänen tai videon reaaliaikaista lähetystä internetissä. Virtaustoistettu media sallii äänen tai videon suoran lähetyksen internetin yli, jolloin internet muuttuu joukkoviestintävälineeksi. Sisältöä voi myös tallentaa virtaustoistettaviin tiedostomuotoihin, jolloin internetin käyttäjät voivat katsoa jälkeenpäin tallenteita tapahtumista.

ERILAISET VIRTASTOISTOTAVAT

On kolme tapaa siirtää ääni- ja videodataa internetin yli: lataaminen ja soittaminen, progressiivinen lataus ja todellinen streamaus.

Lataa ja soita

Lataa ja soita merkitsee sitä, että koko mediatiedosto on ladattava kovalevylle ennen sen soittamista. Tästä syystä lataa ja soita -mediaa ei voida käyttää suoriin lähetyksiin, kuitenkin se on usein hyvä tapa välittää korkealaatuista mediasisältöä millä tahansa aallonleveydellä. Esimerkiksi korkealaatuinen elokuva voidaan ladata minkä tahansa internet-yhteyden yli. Elokuvaa ei kuitenkaan voi toistaa ennen kuin koko tiedosto on ladattu. Niinpä elokuvan pitkä latausaika aiheuttaa ongelmia. DVD-elokuvan (yleensä noin 4.6 GB kokoinen) lataaminen modeemilla (56kbps) kestäisi yli 8 päivää! Useimmat ihmiset pitäisivät tätä aika hitaana!

Lataa ja soita -media ei ole streamaavaa mediaa (enemmän tästä myöhemmin) mutta se on paljon suosittupaa kuin streamaus. Tiedostonjakoverkot ovat lataa ja soita -mekanismeja, ja tämä on erittäin suosittua toimintaa internetissä. Koska laajakaistayhteydet ovat yleistymässä monissa maissa, lataa ja soita -elokuvien jako verkoissa lisääntyy.

Progressiivinen lataus

Progressiivinen lataus merkitsee median soittamista sen latautuessa kovalevylle. Progressiivinen lataus on käyttökelpoinen menetelmä, mutta ei niin hyödyllinen, kuin todellinen streamaus, koska se ei voi esimerkiksi tarjota monen bittitaajuuden koodausta (multiple bitrate encoding). Tämä menetelmä on kuitenkin käyttökelpoisempi kuin lataa ja soita, sillä tiedoston latauksen valmistumista ei tarvitse odottaa ennen kuin sen toistaminen voidaan aloittaa. On olemassa joitain rajoituksia, esimerkiksi jos tiedosto ei lataudu yhtä nopeasti kuin sitä toistetaan, toistoon tulee tauko, kun toistettavaa mediaa ladataan eteenpäin. Tämä voi olla ärsyttävää, minkä vuoksi monet käyttävät mielummin todellista streamausta lähetyksenä.

Quicktime (Applen streamausteknologia) kutsuu progressiivista "Fast Startiksi". Tämä osoittaa yhden ongelman streamauksen opettelemisessa ja teknologioiden erittelyssä, jokainen teknologian tuottaja on luonut oman sanastonsa. Tässä käyttöoppaassa käytetään kuitenkin yleisessä käytössä olevaa terminologiaa, mikäli mahdollista.

Todellinen streamaus

Todellisen streamauksen avulla käyttäjä voi katsoa tai kuunnella mediaa, joka välitetään internetissä reaaliajassa. Todellinen streamaus on ainoa streamaustapa, joka mahdollistaa suoran lähetyksen, ja se tukee myös kehittyneempää toiminnallisuutta kuin progressiivinen lataus, jolloin käyttäjä voi hallita mediakokemustaan pysäyttämällä toiston tai kelaamalla eteenpäin tai taaksepäin.

Todellista streamausta on kutsuttu esimerkiksi verkkolähetykseksi, suoraksi lähetykseksi, verkkotelevisioksi ja nettiradioksi. Vallalla oleva termi on silti "streamaava media" (jota sanotaan

yleensä vain streamaukseksi").

Progressiivisen latauksen ja todellisen streamauksen välinen ero ei aina ole ilmeinen. Jos käyttäjällä on hyvin nopea verkkoyhteys progressiivinen lataus voi näyttää toimivan täsmälleen samalla tavalla kuin todellinen streamaus.

Lataus ja soitto, progressiivinen lataus ja todellinen streamaus jakautuvat kahteen eri luokkaan: staattiseen tiedostostreamaukseen ja suoran lähetyksen streamaukseen.

Staattisen tiedoston streamaus

Ennalta nauhoitettujen mediatiedostojen siirto internetissä reaaliajassa. Kun viittaamme median verkkoarkistoihin, viittaamme staattisten tiedostojen streamaukseen. Suuri videotaiteen arkisto koodattuna streamattaviin tiedostoihin on esimerkki staattisten tiedostojen streamauksesta. Tämä sisältö tunnetaan myös on-demand -sisältönä tai arkistoituna sisällönä. Sisältö voidaan lähettää lataa ja siirrä, progressiivinen lataus, tai todellinen streamaus -prosesseilla.

Streamaus suorana lähetyksenä

Suoran lähetyksen videon tai äänen streamaus internetin yli. Vastaanottaja voi kokea tapahtuman reaaliajassa. Tästä on monia esimerkkejä, esimerkiksi verkkoradiot ja esitysten katsominen suorana lähetyksenä. Vain todellinen streamaus tukee suoraa lähetystä.

Tämä käyttöopas käsittelee lähinnä suoran ja arkistoidun lähetyksen todellista streamausta.

KOODAUS

Staattisten tiedostojen streamaamiseksi verkossa tiedostojen tulee olla pakattuja ja koodattuja "streamausmuotoon". CD-levyjen muuttaminen MP3-tiedostoiksi tekee juuri tämän koodausohjelmilla. Tässä prosessissa CD:n äänitalenne muutetaan koodausohjelmalla CD (Compact Disk) ääniformaatista "streamausääniformaattiin" (MP3-tiedostoksi). Tämä vaatii datan pakkaamista ja sen laadun heikentämistä ja tiedostojen koon supistamista, ja datan muokkaamista "streamausmuotoon" (kuten MP3).

Pakkaus

On kaksi pakkaustapaa - tappioton ja tappiollinen pakkaus.

Kun koodausprosessi pakkaa lähdetiedoston niin, että se voidaan välittää internetyhteyksien yli reaaliajassa, tämä prosessi heikentää äänen ja kuvan laatua. Mitä enemmän tiedostoa pakataan, sitä pienempää kaistaa tarvitaan tiedoston soittamiseen, mutta sitä enemmän laatu laskee. Täytyy saavuttaa kompromissi, jolla saavutetaan hyväksyttävä ääni- ja videokokemus, mutta datan määrää lasketaan tarpeeksi, jotta välitys internetin yli on mahdollista. Suoran lähetyksen vaatima koodaus on samanlainen, paitsi että äänen tai videon (tai molempien) sisääntulo koodataan tiedoston sijasta. Tällä prosessilla koodausohjelma välittää koodatun datan jatkuvana virtana streamauspalvelimelle.

Tappioton pakkaus

Tässä prosessissa datan sisältämä informaatio pakataan pienempään tilaan, ilman että dataa poistetaan. Tämän prosessin voi visualisoida kuvittelemalla paperipussin, jonka sisällä on esine. Kun pussissa oleva ilma poistetaan koko pussin koko supistuu, vaikka pussissa oleva esine ei kutistu.

Tappiollinen pakkaus

Kutsutaan usein näkyväksi koodaukseksi, tämä on prosessi, jossa tietoa "heitetään pois", jotta tiedostokokoa saadaan kutistettua. Pakkausalgoritmit ovat monimutkaisia ja yrittävät säilyttää mahdollisimman hyvän havaittavan laadun, vaikka hylkäävät niin paljon dataa kuin mahdollista.

Tappiollinen pakkaus on tarkkaa työtä. Tämän tekevät algoritmit ottavat huomioon kuinka aivot havaitsevat ääniä ja kuvia, ja sen jälkeen hylkäävät ääni- tai videotiedoston tietoa, vaikka säästävät alkuperäisen lähdemateriaalin äänen ja kuvan kokemuksen. Tämän tehdäkseen prosessit seuraavat psykoakustisia ja psykoviisuaalisia mallinnusperiaatteita.

Kodekit

Ääntä ja videota pakkaavat ja koodaavat ja tiedostomuodot luovat algoritmit tunnetaan "kodekkeina". Sana on tehty kahdesta yleisestä termistä: kompressoidea ja dekompressoidea. Koodausohjelmisto käyttää kodekkia "kompressoimaan" striimin sisällön verkon yli siirtämistä varten, ja sisältöä toistava ohjelma käyttää kodekkia "dekompressoimaan" sisällön toistoa varten.

Kodekki on matemaattista taikuutta ja streamauksen sydän ja sielu. Tästä syystä eri teknologioiden tuottajat taistelevat herruudesta. Jokaisella kodekilla on oma tapansa tehdä asioita, riippuen siitä minkä kehittäjä uskoo olevan paras. Monilla kodekeilla on myös lisensseihin liittyviä ongelmia.

Bittinopeus

Pakkaus on prosessi, jolla datan määrää vähennetään.

Useamman bittinopeuden koodaus antaa palvelimen ja soittimen neuvotella parhaan laadun streamille (korkeimman kaistanleveyden) joka tulee yksittäisestä staattisesta tiedostosta tai suoran lähetyksen streamista. Niinpä soittimeen lähetetään käyttäjän verkkoyhteyden mahdollistaman parhaan mahdollisen laadun streamia. Useamman bittinopeuden koodaus luo vain yhden koodatun streamin.

Bittinopeuksien valitseminen vaatii useamman eri seikan huomioimista. Näihin seikkoihin sisältyy: kohdeyleisön verkkoyhteydet, toivottava videon kuvakoko, liikkeen määrä kuvissa, videon kontrastin taso, nauhoitetun äänen tyyppi (taustahäly, puhe, stereomusiikki), kameraliikkeen määrä, kameran laatu ja kameran linssi.

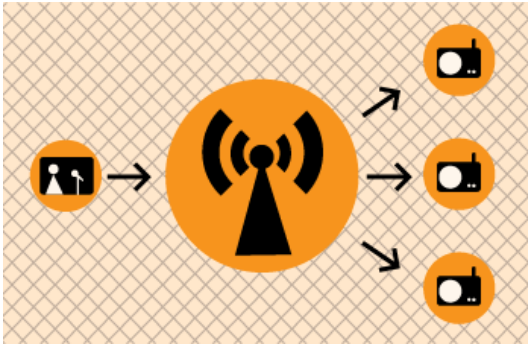
SIIRTO

Staattisten ("arkistoitujen") tiedostojen streamaus saadaan aikaan normaalilla verkkopalvelimella. Tämä on usein paras tapa siirtää sisältöä pienessä mittakaavassa. Tätä menetelmää ei voida käyttää suoriin lähetyksiin eikä se mahdollista todellisen striimauksen kehittyneitä mahdollisuuksia, kuten useamman bittinopeuden koodausta. Tämä menetelmä mahdollistaa myös katkojen ("puskuroinnin") synnyn, eikä se voi tarjota yhtä monia samanaikaisia soitinkytkeä kuin todellinen streamaus.

Mahdollistaakseen suoran lähetyksen striimauksen ja saadakseen täyden toiminnallisuuden ja tehokkuuden vaaditaan sstreamausmediapalvelinta. Tämä palvelin on yleensä tavallinen palvelinlaite, mutta siihen on asennettu tarpeelliset streamauspalvelinohjelmistot. On aika tavallista asentaa streamauspalvelin samalle koneelle kuin verkkopalvelin.

STREAMAUSPALVELIMET

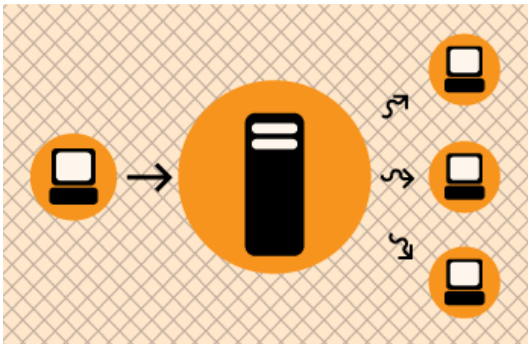
Helppo tapa havainnollistaa streamauspalvelimen toiminta on ajatella radioasemaa. Radioasemassa on kolme osaa - studio, lähetin ja yleisön omistamat vastaanottimet.



Yllä on yksinkertainen kaavio, joka osoittaa kuinka lähettävä radioasema toimii. Radiostudio on äänen lähde. Tässä tilassa on yleensä miksauspöytiä, cd-soittimia, minidisc-soittimia, vinylisoittimia ja niin edelleen. Studiosta tuleva äänisignaali lähetetään lähettimeen. Tämä lähetetään studiosta lähettimeen kaapelia (jota kutsutaan joskus "maalinjaksi") tai mikroaaltolinkkiä pitkin. Lähetin lähettää äänen FM-modulaatiolla, jotta radiovastaanottimet voivat vastaanottaa sen ja toistaa sen.

Radio toimii näin, koska se pyrkii jakamaan studion äänen mahdollisimman monille ihmisille. Jos ajattelemmme radiostudiota ilman lähetintä, asemalla olisi aika rajoittunut yleisö! Käytännössä vain studioon mahtuvat ihmiset voisivat kuunnella. Joten lähetin toimii jakelukanavana, joka antaa useampien ihmisten vastaanottaa radiovastaanottimillaan, ja näin mahdollinen yleisö on paljon suurempi.

Tämä on melko hyvä vertauskuva sille, miksi streamaus on olemassa ja miten se toimii. Jos ääntä soitetaan tietokoneella huoneessa niin yleisö ei tule olemaan suuri... Joten streamausta käytetään jakelemaan ääni useammille ihmisille.



Esimerkki on ilmeinen... Tietokone korvaa radiostudion, streamauspalvelin korvaa lähettimen, ja kuuntelijat kytkeytyvät tietokoneilla palvelimeen sen sijaan, että kytkeytyisivät radiovastaanottimilla lähetettyyn signaaliin. Tätä esimerkkiä voidaan jatkaa pidemmälle. Suuremman radiolähettimen käyttö on kuin suuremman kaistanleveyden hankkiminen streamauspalvelimelle - molemmat antavat useampien ihmisten vastaanottaa ääntä.

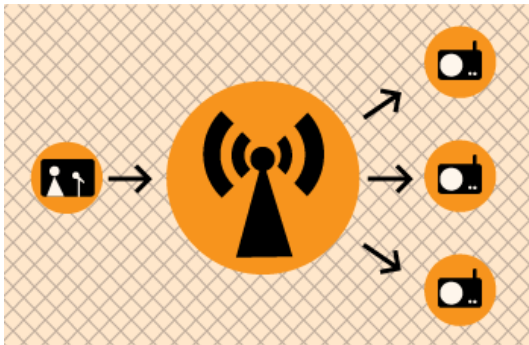
ICECAST



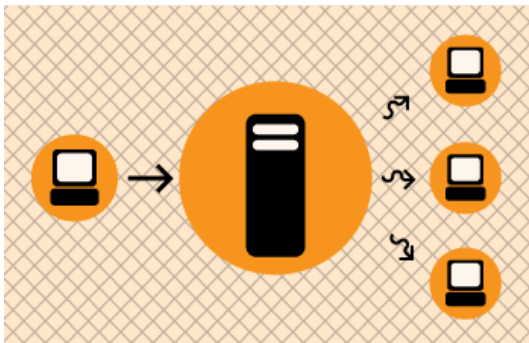
Verkkomaailman audioon ja videoon liittyvät streamaukseksi kutsutut tekniikat. Streamaus on yksinkertaisimmillaan äänen ja videon siirtämistä internetissä reaaliajassa. Perinteisesti streamausta on käytetty suoraan äänen lähettämiseen netissä, tai tapahtumien lähettämiseen internetissä, niin että niitä voi katsella kotona.

Streamauksesta puhutaan usein yleisradiolähetyksen termeillä, sillä internetin suorat ääni- ja videolähetykset ovat usein samankaltaisia yleisradion televisio- ja radiolähetyksen kanssa. Niinpä voit kuulla termin "internet-radio", jolla viitataan suoraan äänilähetykseen netissä. Joskus äänen tai videon suoraan lähetykseen viitataan "radiolähetyksinä internetissä" tai "internet-lähetyksinä". Näistä käsitteistä on apua yritettäessä ymmärtää streamauksen tarkoituksia, mutta niitä ei pidä ottaa liian kirjaimellisesti. On kuitenkin hyödyllistä ajatella näitä malleja, kun Icecastin roolia pohditaan.

Televisio- ja radiolähetyksiä varten on lähetin, joka lähettää studiosta lähetetyn signaalin televisioosi tai radioosi.



Internet-lähetyksiä (streamausta) varten korvaat lähettimen streamauspalvelimella, joka on itse asiassa ohjelmisto.



Icecast on tällainen ohjelma. Se mahdollistaa suoran ääni- ja videolähetyksen netissä reaaliajassa. Huomaa, että tämä on vain osa yhtälöstä. Tarvitset myös streamin koodaajan, joka voi lähettää alkuperäisen ääni- tai videostreamin Icecastiin. Icecast jakelee tämän streamin, se ei luo sitä - se on koodaajan tehtävä.

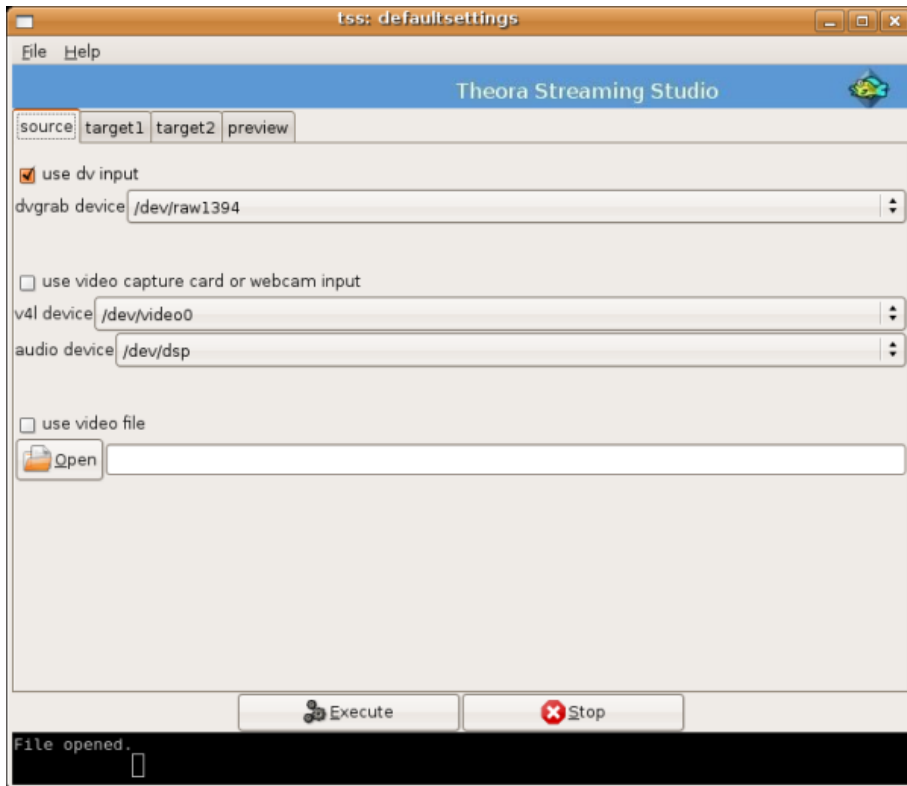
Icecast on hyvin kypsä ja kestävä teknologia. Se tukee monien streamaavan äänen formaattien lähettämistä (sisältäen MP3 ja Ogg Vorbis -formaattit) ja se tukee Ogg Theoran lähettämistä streamaavana videona. Icecast toimii Linuxissa ja Windowsissa. Jos tahdot asentaa sen Linuxiin, sinulla täytyy olla jonkin verran kokemusta Linuxin kanssa, erityisesti tulee tuntea ohjelmien asentaminen komentoriviltä, ja kyetä editoimaan tekstiin perustuvia konfiguraatitiedostoja. Windowsin asennusprojekti on suoraviivaisempi, mutta silti täytyy kyetä editoimaan tekstipohjaisia konfiguraatitiedostoja.

Icecastin asentaminen ja käyttäminen ei sovi kokemattomille käyttäjille.

TSS

TSS (Theora Streaming Studio) on erinomainen koodaaja Ogg Theora -streamien lähettämiseen Iccast-serverille. TSS-kotisivut kannattaa lukea, sillä niillä on yleiskatsaus sen toimintoihin ja asennusohjelmia:

<http://gollum.artefacte.org/tss/>



TSS toimii graafisen käyttöliittymän avulla, joten ei ole tarpeen tehdä mitään komentorivillä. TSS voi lähettää yhden tai useampia streameja eri asetuksilla samasta videolähteestä. Videolähde voi olla digitaalinen video tai analoginen video videokaappauslaitteen kautta, jollainen voi olla verkkokamera, USB-videosisääntulo tai video-in-liitäntä. Pidä mielessäsi, että nopea tietokone, jossa on ainakin kaksi gigaa RAM-muistia tarvitaan streamaamaan kahteen kohteeseen ilman freimien menetyksiä tai audiosynkronisaation menetystä.

TSS voi myös näyttää ulos menevän streamin esikatselun, se voi samaan aikaan arkistoida streamit levykkeelle (mutta tämä riippuu hieman siitä, kuinka nopea tietokoneesi on).

TSS hallinnoi komentorivin ohjelmia:

1. **dvgrab** - kaappaa videon suorasta lähteestä, kuten kamerasta
2. **ffmpeg2theora** - muuntaa videotiedostot Ogg Theoraksi
3. **oggfwd** - lähettää videot streamauspalvelimeen

TSS ottaa graafisessa käyttöliittymässsä asettamasi konfiguraation ja luo asiaankuuluvan komentorivin käyttäen näitä kolmea ohjelmaa, jotta et joudu asettamaan niitä monimutkaissa komennoissa käsin.

Voidaan aina kopioida kometorivi, jota TSS käyttää, se printataan statusikkunaan, pieneen mustaa tilaan, joka näkyy **Execute** ja **Stop** -napuloiden alla. Vieritä hiiren osoittimella ja valitse teksti. Voit sen jälkeen ajaa nämä komennot suoraan komentoriviltä, säästäten joitain resursseja välttämällä graafisen järjestelmän avaamisen.

TSS:n pääkehittäjä on Lluís Gómez i Bigordà, nykyinen julkaisu (helmikuu 2009) on 0.2. Projektin status on aktiivinen.

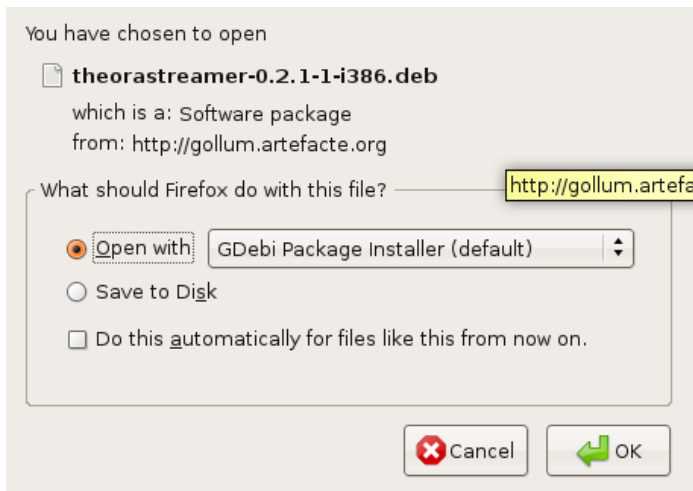
TSS:N ASENNUS

TSS voidaan asentaa vain Linuxiin. Katsomme nyt TSS:n asennusta Ubuntuissa.

Käy ensin TSS-lataussivulla:

<http://gollum.artefacte.org/tss/#download>

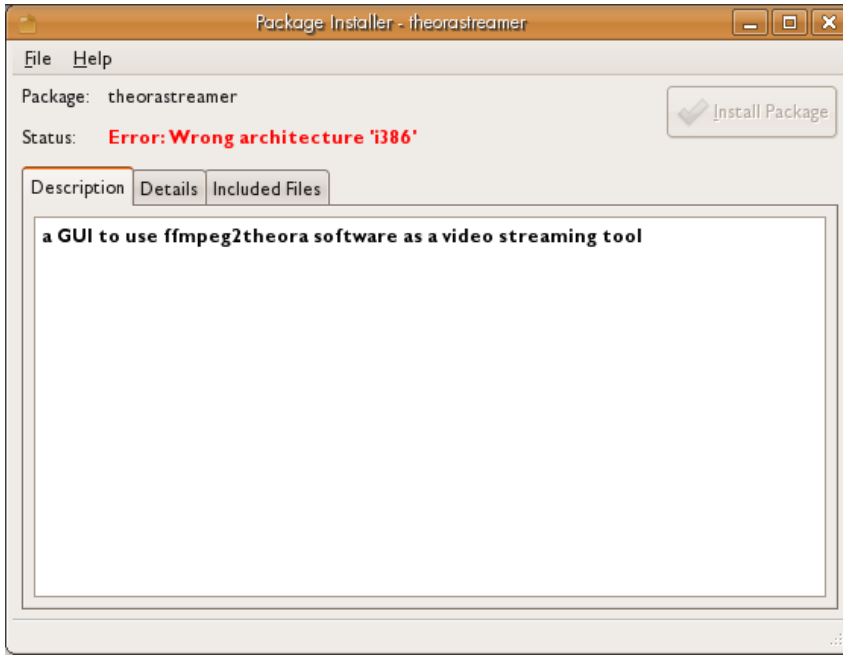
Näet osan, jossa on kaksi linkkiä. Yksi linkki on "Debian / Ubuntu package". Tahdomme asentaa sen version. Napsauta tätä linkkiä käyttäen FireFoxia, ja näemme tällaisen ikkunan ilmestyvän:



Optio on oletusarvoisesti juuri se, jota tahdomme käyttää. Jos avaa asennusohjelman **GDebi-paketinasennusohjelmalla**, silloin asennusohjelma lataa ja asentaa automaattisesti. Tehdään niin. Klikkaa ja se asentuu.

Asennus 64-bittiseen

Kun asennat TSS-ohjelmaa, voit nähdä tällaisen virheen:



Tämä merkitsee, että yrität asentaa koneen tietokoneelle, jonka "arkkitehtuuria" ohjelma ei salli. Voit kiertää tämän, mutta sinun täytyy tehdä se komentorivillä. Ensinnäkin, GDebin käytön sijasta joudut lataamaan lähdetiedostot (.tar.gz -tiedosto). Sitten joudut kirjoittamaan tämän komentorivillä samasta hakemistosta, jossa asennusohjelma on:

```
sudo apt-get install dvgrab ffmpeg ffmpeg2theora libtheora-bin oggfwf libvte-dev
```

Edellinen komento voi kestää hetken, sillä on paljon ladattavaa ja asennettavaa. Kun prosessi on valmis, kirjoita nämä komennot:

```
tar zxvf tss-0.2.tar.gz
cd tss-0.2
./autogen.sh
```

Huomaa: asennettavan tiedoston nimi on luultavasti erilainen kuin yllä.

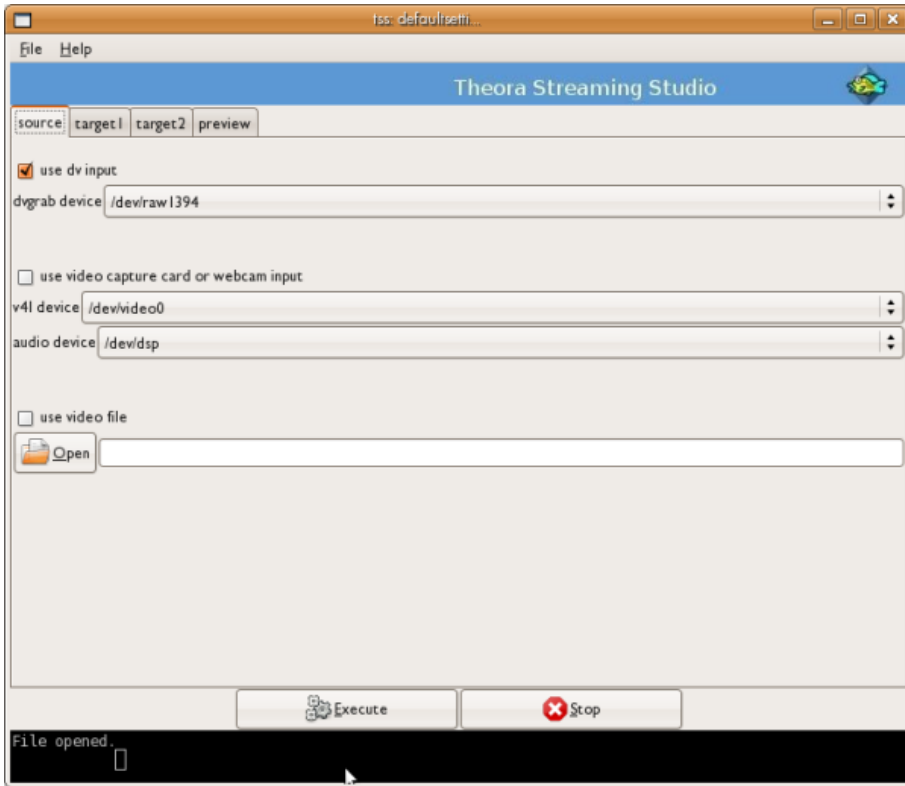
Sen jälkeen aja seuraavat komennot:

```
make
sudo make install
```

Kirjoita "TSS" komentorivillä. Jos se ei toimi, kokeile tätä:

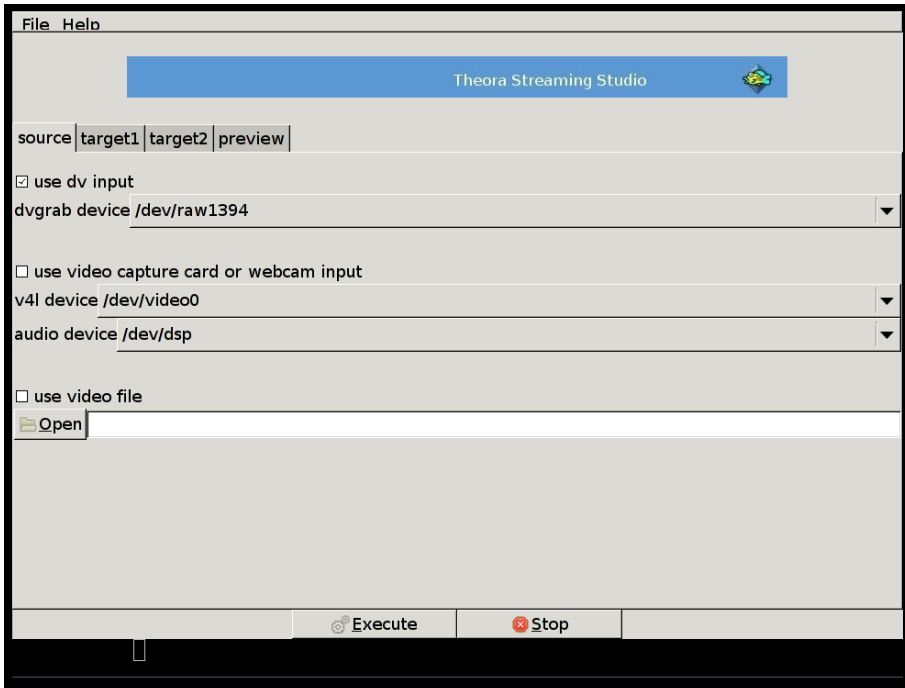
```
sudo cp src/tss /usr/bin/
```

Kokeile sitten aloittaa TSS uudestaan komentoriviltä:



TSS KÄYTÖSSÄ

On kaksi erilaista pääpaneelia, joiden täytyy sisältää oikeat asetukset, jotta striimiä voi lähettää: lähde ja kohde -välilehdet:



Source -välillehdestä voit valita videon sisääntulon tyyppin. Kun käytät minimaalista järjestelmää, on olemassa vaihtoehto käyttää verkkokameraa video for linuxin (v4l) kautta. Voit valita myös digitaalisen videon sisääntulon - tämä on hyödyllinen, jos tahdot lähettää suoraa videota kädessä pideltävästä kamerasta.

TSS tarjoaa myös mahdollisuuden streamata olemassaoleva (arkistoitua) videotiedosto tietokoneeltasi Iccast-palvelimelle.

Toinen välilehti on **target1**, tässä määrittelemme streamien asetukset päästreamiamme varten. Alla on esimerkki:

The screenshot shows the Theora Streaming Studio (TSS) interface. The top bar has 'File' and 'Help' menus. Below is a blue header with 'Theora Streaming Studio' and a logo. The main area is divided into tabs: 'source', 'target1', 'target2', and 'preview'. The 'target1' tab is active. It contains four main sections: 'ffmpeg2theora encoding settings', 'ffmpeg2theora metadata', 'icecast server settings', and 'icecast metadata'. The 'encoding settings' section has dropdown menus for frame size (320x240), framerate (25:3), video quality (3), audio quality (2), audio channels (1), and audio samplerate (22050). The 'metadata' sections have text input fields for artist, title, date, location, organization, copyright, license, description, genre, name, and URL. The 'icecast server settings' section has text input fields for server, port, mountpoint, and passwd. The 'icecast metadata' section has text input fields for description, genre, name, and URL. At the bottom, there are buttons for 'Execute' and 'Stop', and a 'dump to local file' section with a 'Browse' button.

Kaikki kentät kohdassa **ffmpeg2theora encoding settings** pitää täyttää.

On tärkeää pitää mielessä, että kaikella tähän laittamamme tieto vaikuttaa suoraan streamien lopulliseen ulostulon bittinopeuteen. et voi vain laittaa ulostulon bittinopeutta, pikemminkin se riippuu ruudun koon, äänen laadun ja muiden tekemiesi valintojen asetuksista.

Muista myös, että mitä korkeampi tämä bittinopeus on - sitä enemmän kaistanleveyttä tarvitaan sen lähettämiseen Iccast-palvelimelle, ja sitä korkeammat ovat streamiisi kytkeytyvän yleisön kaistanleveysvaatimukset. Joten, jos osa katsojistamme on Kolumbiassa, jossa keskimääräinen verkkoyhteys on vieläkin noin 128kbps (**kb/s** tai **kbit/s** on tiedonsiirron nopeusyksikkö, joka vastaa noin 1000 bittiä sekunnissa). Tämä merkitsee, että vaikka meillä on laajakaistayhteys, joka riittää streamin lähettämiseen korkealla resoluutiolla ja korkeatasoisella pakkauksella, meidän täytyy varoa yli 128kbps bittinopeuden käyttöä, koska muuten ihmiset Kolumbiassa eivät näe sitä.

Seuraavat kategoriat liittyvät TSS-enkoodauksen vaihtoehtoihin sekä **target1** että **target2** -kohteille:

Frame size (ruudun koko): Tässä määrittelemme videostriimin ulottuvuudet pikseleinä. 320 kertaa 240 on standardi. Alhaiselle kaistanleveydelle voidaan käyttää ruudun kokoa 320 kertaa 180.

Frame rate (ruutunopeus): Tämä asetus määrittelee, kuinka monta ruutua voimme lähettää sekunnissa. Erilaiset tapahtumat vaativat erilaisia ruututaajuuksia. Konferenssin tapauksessa kohteet eivät liiku tai vaihda paikkaa kovin paljon, joten 12 fps (fps = frames per second, ruutua sekunnissa) -asetus voi olla sopia. Edellisessä esimerkissä päätimme uhrata ruututaajuuden suuremman ruudun koon hyväksi. Valitsimme 25:3, mikä merkitsee noin 8fps ruututaajuutta.

Video Quality (videon laatu): Tähän asetukseen voi laittaa arvoja väliltä 0-10. Oletusarvoinen videon enkoodauslaatu ffmpeg2theoralle on 5, suuremmat arvot merkitsevät parempaa laatua.

Audio Quality (äänen laatu): Tähän asetukseen voi laittaa arvoja väliltä -2 - 10. Enkoodauslaatu äänelle on oletusarvoisesti 1. Suuremmat arvot merkitsevät parempaa laatua.

Audio Channels (äänikanavat): Asettaa ulostulon äänikanavien määrän. Valinnat ovat 0 (ei ääntä), 1 (mono) ja 2 (stereo).

Audio samplerate (äänen näytetaajuus): Esitetään Hertzeinä. Tämä laatikko asettaa äänen ulostulon näytetaajuuden.

Ulosmenevän streamin teknisiä asetuksia käsittelevien laatikoiden alle voit kirjoittaa tiedot, jotka tahdot yleisön tietävän:

ffmpeg2theora metadata	
artist	put your name here
title	the title of the show
date	5-5-2007
location	Marrackesh
organization	artefacte
copyright	artefacte
license	creative commons share alike

Tämä informaatio **ffmpeg2theora metadata** -laatikoissa on potentiaalisesti tarjolla yleisöllesi. Riippuu heidän tavastaan katsella streamia (selaimella tai soittimella), näkevätkö he tämän metadataa. Voit kirjoittaa tähän mitä tahansa tietoa, sillä ei ole väliä - jätä huomiotta kategoriat (sijainti, päivämäärä jne.), joita et tahdo käyttää.

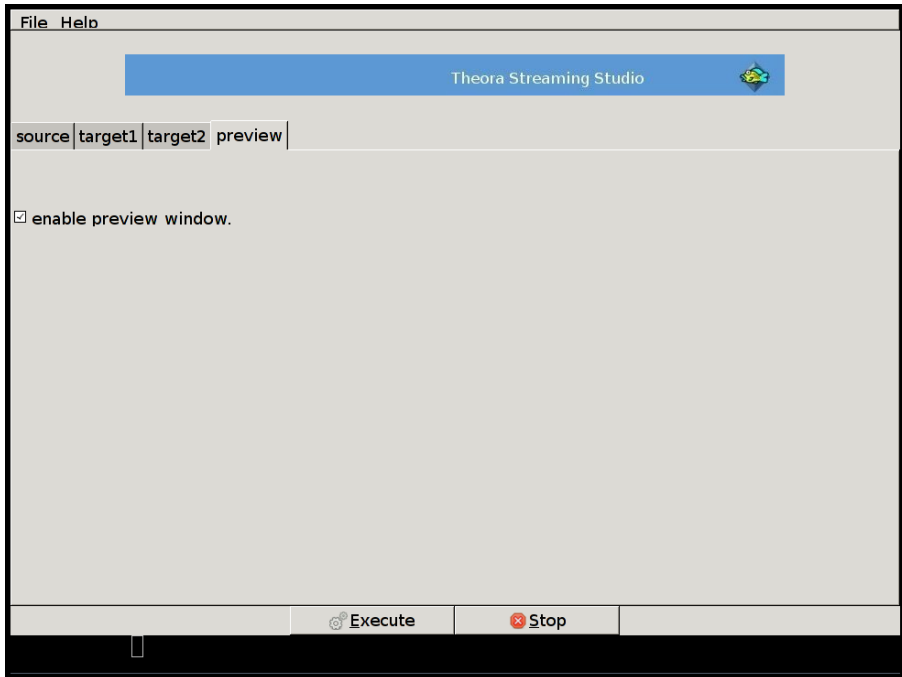
Seuraavaksi on laatikot **icecast server settings** ja **icecast metadata**.

icecast server settings	
server	gollum.artefacte.org
port	8000
mountpoint	/yourmountpointthere.ogg
passwd	
icecast metadata	
description	between love and coding
genre	underground burlesque
name	the title of the show
URL	http://gollum.artefacte.org/tss

Kohdassa **icecast server settings** meidän täytyy määritellä streamauspalvelimen osoite (server), tämän serverin käyttämä portti (**port**), joka on yleensä 8000, käyttämämme kanava (**mountpoint**), ja salasana (**password**). Jos sinulta puuttuu pääsy Icecast-palvelimelle, on olemassa muutamia ilmaisia projekteja, jotka pyrkivät auttamaan ihmisiä vapaiden ohjelmistojen käytössä. Katso estudiolivres.org tai GISS.tv.

Icecast metadata -asetuksiin voit laittaa yleisölle hyödyllistä tietoa streamistasi. Tämä tieto näytetään palvelimen verkkosivulla (jokaisella Icecast-palvelimella on oma verkkosivunsa). Vaikka niillä on omat otsikkonsa, voit laittaa tähän mitä tahansa tietoa, sillä ei ole väliä.

Viimeisessä välilehdessä on otsikon esikatselu (**preview**).



Jos rastitat kohdan **enable preview window** (esikatselu-ikkuna päällä), näet esikatselun suorasta lähetyksestä enkoodatessasi sitä. Tämä ei ole sama video kuin se, joka lähtee Icecast-palvelimelta, joten kannattaa aina katsoa sekä esikatselua että palvelimelta tulevaa streamia (monitoroi sisääntulevaa streamia toisella tietokoneella).

Jos tahdot lähettää yhden streamin, joudut katsomaan vain asetuksia välilehdissä Source ja Target1. Jos tahdot lähettää kaksi streamia, joudut katsomaan asetuksia myös välilehdessä Target2.

Kun olet määritellyt kaikki tarvittavat tiedot jokaiselle ulosmenevälle streamille ja yhdistänyt oikean videolähteen (tai olet valinnut toimivan videotiedoston streamattavaksi), paina vain "Execute" ja streamauksen pitäisi alkaa!

FFMPEG2THEORA-STREAMAUS

Tämä menetelmä keskittyy videon streamaukseen digitaaliselta videokameralta tai Firewire-kameralta GNU/Linuxia käyttäen. Videostreami on Ogg Theoraa, joten katsojasi voivat nähdä sen käyttäen Theora-yhteensopivaa soitinta, kuten FireFox 3.5, tai Java-sovelmaa, kuten Cortado. Tätä seurataksesi sinun täytyy osata käyttää komentoriviä ja sinulla täytyy olla pääsy sudo-komennon käyttöön. Jos et tiedä mitä nämä asiat ovat, kannattaa lukea hyvä kirja komentorivin käytöstä.

VALMISTAUTUMINEN

Huomaa: **Firewire**-kamera ei ole sama kuin digitaalinen videokamera. Kuitenkin tässä tapauksessa ne toimivat samoin. Firewire on Applen nimi **IEEE 1394** -standardille, muuta valmistajat käyttävät esimerkiksi nimiä i.Link (Sony) tai Lynx (Texas Instruments) - mutta sillä ei ole väliä, kaikki ovat standardin IEEE 1394 mukaisia.

Tarvitset seuraavat:

- DV-kamera
- GNU/Linux-kone, jossa on Firewire-sisääntulot
- Firewire-kaapeli
- Internet-yhteys
- Pääsy "theora-yhteensopivalle" Iccast2-palvelimelle (tarvitset palvelimen **IP-osoitteen**, käytettävän **portin**, ja **salasanan**)

ASETUKSET

Seuraavaksi asennetaan tarvittavat ohjelmat. Meidän täytyy asentaa seuraavat ohjelmat:

- dvgrab
- ffmpeg2theora
- oggfwf

Ubuntussa voit ajaa tämän komentorivin:

```
sudo apt-get install dvgrab ffmpeg2theora oggfwf
```

Edellisen komennon pitäisi näkyä yhdellä rivillä. Sinulta kysytään salasanaa, kirjoita tämä ja asennusprosessi alkaa. Nyt sinulla on kaikki mitä tarvitset aloittamiseen. Ensin meidän täytyy yhdistää digitaalinen videokamera tai Firewire-kamera tietokoneeseen. Sinun täytyy kytkeä Firewire-kameraan Firewire-kaapeli, ja toinen kaapeli tietokoneesi Firewire-sisääntuloon.

Käännä nyt kamera päälle.

Seuraavaksi sinun täytyy kirjoittaa seuraava komentorivi, joka tarjoaa streamauspalvelimen yksityiskohdat Iccast2-palvelimelle. Komento on:

```
sudo dvgrab --format raw - | ffmpeg2theora -a 0 -v 5 -f dv -x 320 -y 240 -o /dev/stdout - | oggfwf iccastserver 8000 pwd /theora.ogv
```

Muista, että komennon täytyy olla kokonaan yhdellä rivillä (edellinen esimerkki ei ole). Korvaa myös alla olevat yksityiskohdat tiedoilla, joita sinulla on Iccast2-palvelimesta:

- *iccastserver*
- *8000*
- *pwd*

icecastserver korvataan streamauspalvelimen **nimellä** tai **IP-osoitteella**. **8000** on portin numero ja on yleensä sama. Salasanaksi kirjoita palvelimesi salasana. Lopulta voit korvata kohdan **/theora.ogv** asennuspisteelläsi, tämän riippuu Icecast-palvelimesi asetuksista, mutta voi olla mitä tahansa, kunhan se alkaa eteenpäin kallistuvalla kenoviivalla (/) ja loppuu **.ogg** tai **.ogv**.

Nyt sinun pitäisi voida streamata! Tarkastaaksesi yhteyden, kokeile sitä ohjelmilla **VLC** tai **Firefox 3.5**.

SUORATOISTO VLC-OHJELMALLA

VLC voi myös suoratoistaa Theora-tiedostoja **Icecast2** -streamauspalvelimelle. Tämä toimii VLC versiosta 0.9.4 ylöspäin **Ubuntussa**, ja myös VLC:n aiemmilla versioilla Windowsissa (tosin prosessi on erilainen VLC:n aiemmissa versioissa). On mahdollista streamata videotiedostoja koneeltasi, elävää verkkokameran kuvaa tai käsin pideltä kameraa. Jos sinulla on sylimikro, jossa on sisäänrakennettu kamera, voit streamata myös sitä käyttäen.

Jos käytät Ubuntuä, on parempi päivittää Ubuntun versio Intrepidiiin (8.10) tai Jauntyyn (9.04), koska molemmissa on VLC:n versio, jolla on helppo streamata. Jos et tahdo päivittää käyttöjärjestelmää, joudut asentamaan VLC:n viimeisimmän version lähdetiedostosta - ole varovainen, VLC:n päivittäminen lähdetiedostoista on vaikeaa.

SUORATOISTOPALVELIMEN VAATIMUKSET

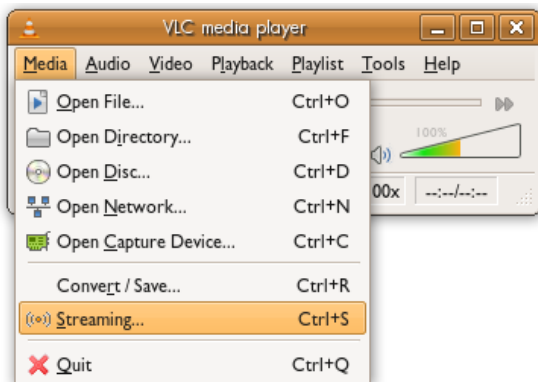
Tutkimme striimausta **Icecast-palvelimelle**. Sinulla täytyy olla yhteys joko Icecast-palvelimelle tai johonkin, joka voi luoda sinulle käyttäjätilin ja tarjota pääsy tiedot. Sinun täytyy tietää seuraavat seikat Icecast-palvelimesta:

- **Verkkonimi tai IP-osoite**
- **Portti** (Oletusarvo on **8000**)
- **Salasana**
- **Käyttäjänimi** (Tämä ei välttämättä ole tarpeen riippuen **Icecastin** asetuksista.)
- **Mountpoint** (Tämä ei välttämättä ole tarpeen riippuen **Icecastin** asetuksista.)

STREAMATTAVAN VIDEOTIEDOSTON VALINTA

Katsotaanpa streamausta käyttäen kannettavaan koneeseen sisäänrakennettua kameraa. Prosessi on hyvin samanlainen kuin käytettäessä ulkoista kameraa tai arkistoituja videotiedostoja.

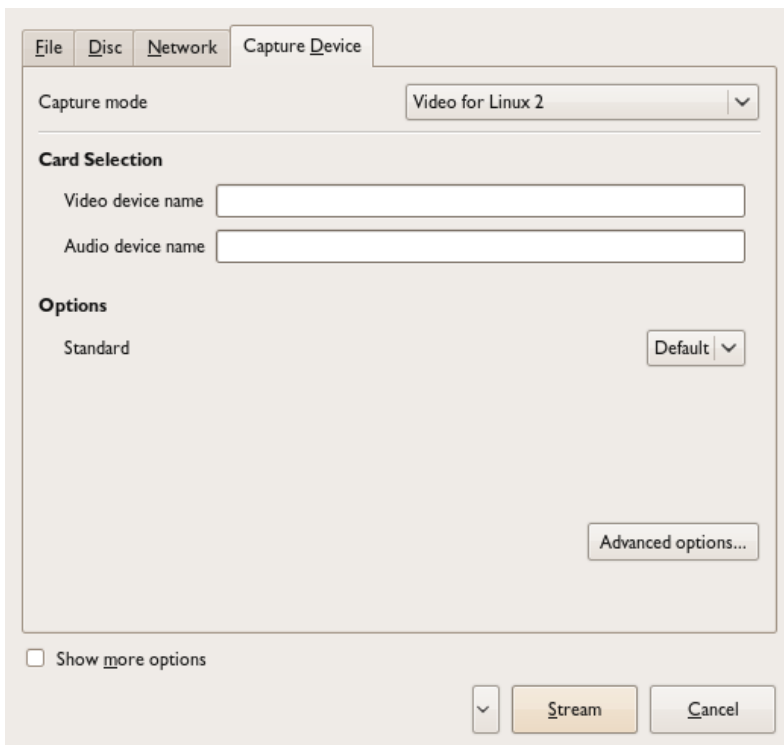
Ensin sinun täytyy avata **VLC**. Valitse nyt Streaming valikosta **Media**:



Tämä avaa paneelin, josta voit valita videostreamin lähteen.



Valitsemme viimeisen välilehden - **Capture Device**, tässä voit kuitenkin valita toisen välilehden streamataksesi videota levykkeeltä (esimerkiksi DVD:ltä), arkistoiduista tiedostoista, tai välittää sisään tulevan videostreamin. Kaappauslaitteen ikkuna näyttää tälle:



Jos kaikki toimii hyvin, sinun ei ehkä tarvitse muuttaa mitään asetuksia tässä ikkunassa. Sen sijaan napsauta vain Stream-nappulaa.

STREAMIN ASETUKSET

Streamin asetusten Outputs-kohdassa on osio Icestadin konfiguroimiseen.

☐ IceCast	Address	Port: 1024
	Mount Point	Login/pass:

Sinun täytyy vain rastia laatikko ja täyttää yksityiskohdat. Tässä on muutama asia muistettavaksi:

1. Mountpoint alkaa aina merkillä "/".
2. Osoite ei ala "http://".
3. Yleensä et saa käyttäjänimeä Icestadiin. Tässä tapauksessa käyttäjänimi on "source".
4. Portti on melkein aina 8000.
5. Muoto kohdalle **Login:pass** on aivan kuin otsikko vihjaa - login, jota seuraa ":", sitten pass. Kaikki ilman välilyöntejä. Esimerkiksi striimi Icast2:n oletusarvoiseen asennukseen (oletusarvoinen salasana on aina "letmein") olisi:

source:letmein

☒ IceCast	Address icecast.mysite.com	Port: 8000
	Mount Point /live.ogg	Login/pass: source:letmein

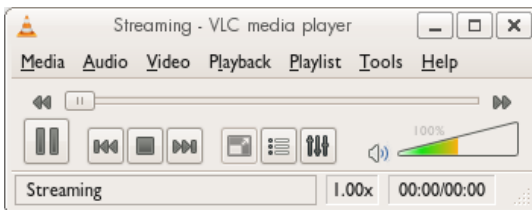
Kun kirjoitat tiedot, näet tekstin muuttuvan kohdassa 'Generated stream output field' ikkunan pohjalla.

Generated stream output string

`:sout=#duplicate{dst=std{access=shout,mux=ogg,dst=source:letmein@icecast.mysite.com:8000/live.ogg}}`

Kokenut käyttäjä voisi kirjoittaa tämän tiedon suoraan käyttämättä tarjottuja kenttiä, mutta on paljon helpompaa täyttää laatikot, kuten vaaditaan.

Nyt sinun täytyy vain painaa 'Stream' ja suoran lähetyksen streami lähtee Icast2 -palvelimelle. Kuitenkaan VLC ei tarjoa kunnon virheilmoituksia, joten se voi istua aloillaan näyttäen sille, että streamaa onnessaan, vaikka todellisuudessa mitään ei tapahdu. VLC voi jopa näyttää "Streaming" statuspalkissa, mutta tämä ei merkitse mitään!



Ainoa tapa saada selville toimiiko se, on katsoa Icast-palvelimesi statussivua, tai käyttää videosoitinta toistamaan streami suoraan palvelimelta.

THEORA-VIRTAUSTOISTON TALLENTAMINEN

Jos käytät GNU/Linuxia, on helppoa tallentaa Ogg Theora -striimi. Tämä johtuu siitä, että Theora käyttää HTTP:tä virtaustoistoon, joten voit käyttää **wget**-ohjelmaa, joka on yleensä mukana Linux-distroissa. Käytät **wget**-komentoa virtaustoiston kaappaamiseen näin:

```
wget http://streamingserver:8000/streamname
```

Esimerkiksi, jos palvelin on osoitteessa **icecast.streaminguitcase.com** ja virtaustoistotiedoston nimi on **theora.ogg**, voit käyttää seuraavaa komentoa arkistointiin:

```
wget http://icecast.streaminguitcase.com:8000/theora.ogg
```

Tämä on hyvä, mutta jos **wget** puskuroituu, se lopettaa arkistoinnin. Voit kuitenkin kokeilla tätä:

```
while true; do wget http://icecast.streaminguitcase.com:8000/theora.ogg;done
```

Toinen etu **wget**-komennon käytöstä arkistointiin on, että se lisää tiedostonimet. Jos arkistoit striimin yllä mainitun esimerkin mukaan, tiedosto tallennetaan nimellä **theora.ogg**. Jos pysäytät **wget**-komennon ja aloitat sen uudelleen, seuraavan tiedoston nimi on **theora.ogg.1**. Tämä merkitsee, ettet ylikirjoita edellistä arkistointitiedostoa.

VIDEON OSIEN EKSTRAKTOIMINEN

Videoita, jotka eivät ole suorana lähetyksenä, on joskus mahdollista ekstraktoida osittain. Tämä toimii vain, jos palvelimen ylläpitäjällä on **oggz-chop** asennettuna. Se sallii hakemisen palvelinpuolelta. Jos se on installoitu, voit tarkentaa "sisään" ja "ulos" -kohdat, joten jos tahdot vain ladata videon 23 sekunnin kohdalta 42 sekunnin kohdalle, lisäät `?t=23/42` URLin loppuun:

```
wget http://example.org/theora.ogv?t=23.0/42.0
```

Firefox 3.5 -selaimella voit myös avata URLin selaimessa, ja tallentaa sen napsautamalla hiiren oikealla napilla Save Video As...

EDITOINTI

30. JOHDANTO

31. PITIVI

JOHDANTO

Ogg Theora on loistava jakeluformaatti, mutta sitä voidaan myös editoida. Jos sinulla on GNU/Linux-kone, jossa on tarpeeksi muistia ja laskentatehoa, voit tutkia editointiohjelmia, kuten **PiTiVi** (<http://www.pitivi.org/>), **Kdenlive** (<http://www.kdenlive.org/>) ja **Cinelerra** (<http://cinelerra.org/>).

Tässä vaiheessa PiTiVi on paras vaihtoehto Theora-tiedostojen yksinkertaiseen editointiin. Kdenlive kaatuu usein, kun tiedostot viedään Theoraan (ja ulostuleva tiedosto on väärin formatoitu) ja Cinelerraa on vaikea sekä asentaa että käyttää.

Jos käytät Ubuntuä, PiTiVi on vielä parempi vaihtoehto, sillä asennus on hyvin yksikertaista. Lisäksi PiTiVi on rakennettu **GStreamer** -multimediakehityksen ympärille (jota Ubuntu käyttää), joten voit tuoda ja editoida miltei mitä tahansa GStreamerin tukemaa mediatiedostoa (joita on paljon), ja viedä sen Theoraan.

PiTiVi toimii kuin useimmat editointiohjelmat: se purkaa videotiedostot prosessissa, jossa videodata tuodaan editointiohjelmaan joten voit katsella ja selata kaikkia kuvia ja ääniä, järjestää ne uudelleen, ja tehdä kaikenlaisia juttuja. Kun lopetat, pakkaat editoidun materiaalin viemällä tuloksen. Tällä pakkaamisen ja purkamisen prosessilla on kuitenkin hintansa, sillä menetät hieman laatua jokaisella pakkauksella tai purkamisella, jonka teet. Tämän vuoksi editointiprosessia kutsutaan tappiolliseksi editoinniksi.

PITIVI

PiTiVi on videoeditori GNU/Linux -käyttöjärjestelmään. Se tukee Ogg Theora -videon tuomista ja viemistä. Juuri nyt se tukee peruseditointia, videoleikkeiden leikkaamista, aikajanelle järjestämistä ja äänitasojen säätämistä.

PiTiVillä on oma dokumentaationsa, jota tahdot ehkä lukea. Se löytyy osoitteesta <http://www.pitivi.org/wiki/Manual#Proposed>

Erittäin hyvä PDF-muodossa oleva manuaali on myös saatavilla osoitteesta <http://jeff.ecchi.ca/blog/?p=897>

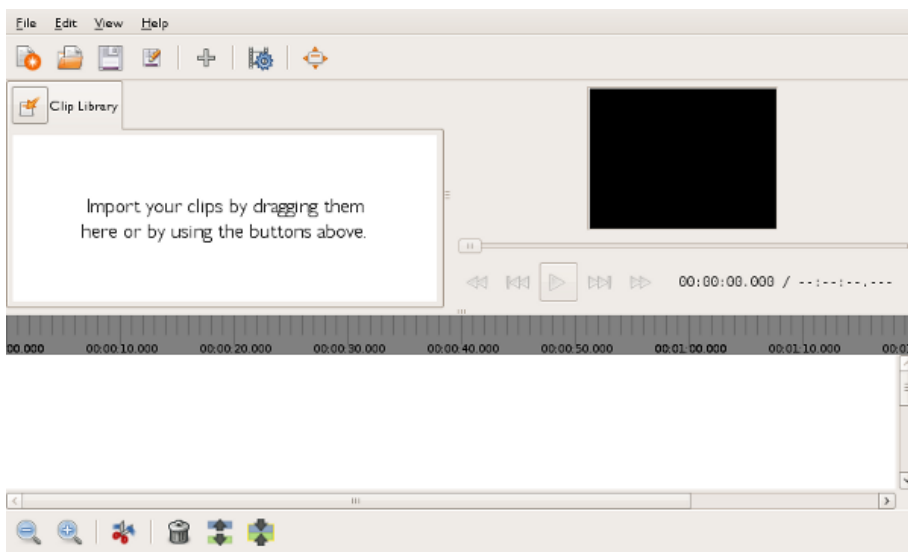
ASENTAMINEN

PiTiVi toimii vain GNU/Linux -käyttöjärjestelmässä. Jos sinulla on Ubuntu Jaunty (9.04), voit asentaa erittäin käyttökelpoisen version PiTiVistä komentoriviltä (pääteestä) komennolla:

```
sudo apt-get install pitivi
```

Sinulta kysytään salasanaa, sen kirjoitettuaasi asennus tapahtuu automaattisesti.

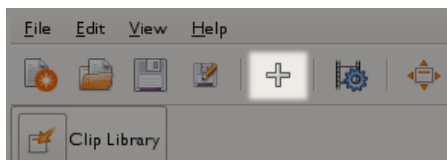
Kun asennus on valmis, voit avata PiTiVin Sovellukset-valikosta (listattu kohdassa **Ääni & Video**).



Voit myös katsoa linkistä <http://pitivi.org/wiki/Downloads> ohjeita uudempien versioiden asennukseen. PiTiViä kehitetään juuri nyt aktiivisesti, uusia versioita tulee säännöllisesti.

VIDEON TUOMINEN

Kun olet avannut PiTiVin, voit painaa plus (+) näppäintä käyttöliittymässä, jotta voit Tuoda leikkeitä.



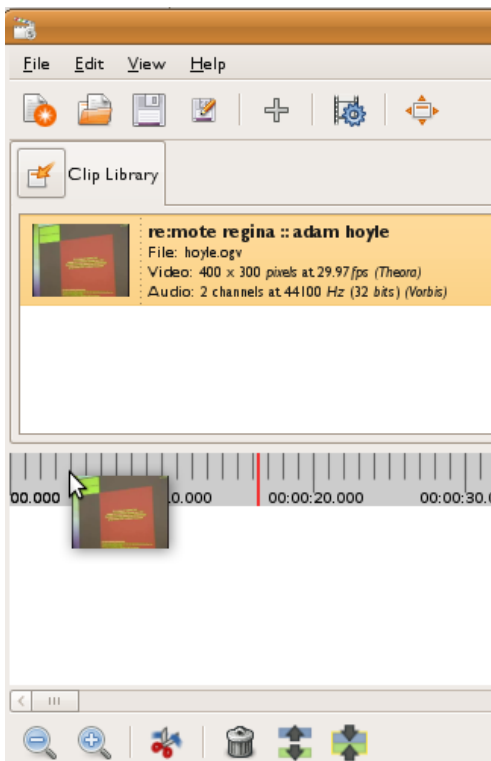
Tuodut viedoleikkeet on lisätty **Clip Libraryyn**. Voit käyttää kaikkia formaatteja, joita **GStreamer** tukee. Jos et tiedä mitä tämä tarkoittaa, helpoin tapa tietää mitkä tiedostot toimivat PiTiVillä on soittaa niitä Totemissa (löytyy Ääni ja Video -sovellusvalikosta ja on nimetty "Elokuvasoittimeksi"). Jos tiedosto toimii Elokuvasoittimessa, sitä voidaan käyttää myös PiTiVin viedoleikkeenä.

Voit jatkaa leikkeiden lisäämistä Clip Libraryyn (digitaalisen kellon videon on luonut Riccardo laconelli). Niitä voi lisätä niin paljon kuin haluat.

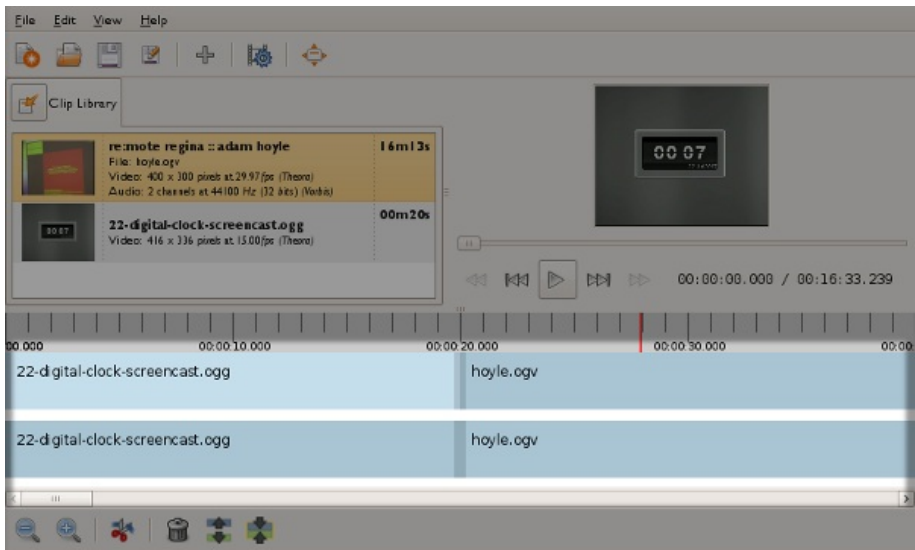


VIDEOLEIKKEIDEN LISÄÄMINEN

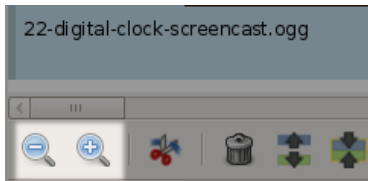
Kun olet tuonut viedoleikkeet, joita tahdot käyttää, voit viedä ne Clip Librarystä alla olevalle aikajanalle.



Jos lisäät monta viedoleikettä aikajanalle, ne näkyvät hieman erilaisissa väreissä ja tiedostonimet näkyvillä:

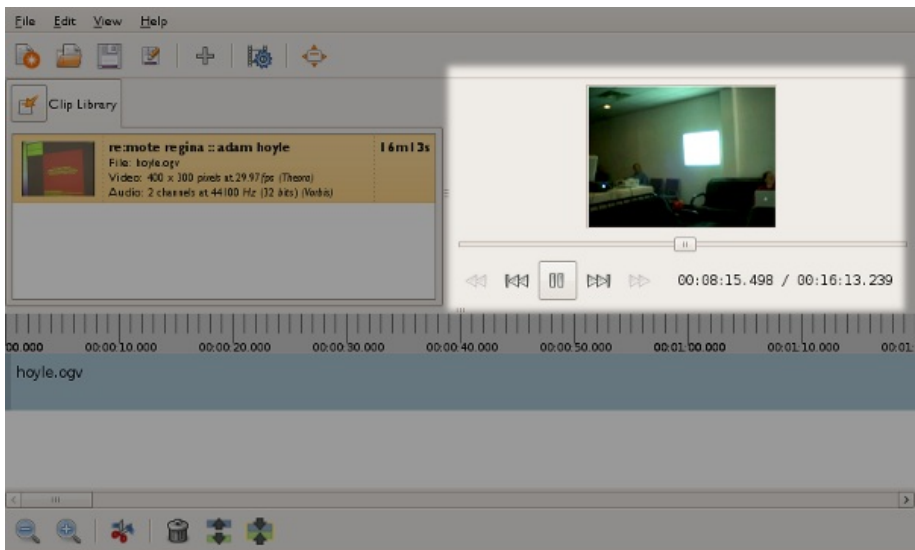


Jos et näe tällaisia yksityiskohtia, käytä zoomaustyökaluja lähentääksesi ja loitontaaksesi aikajanaa:

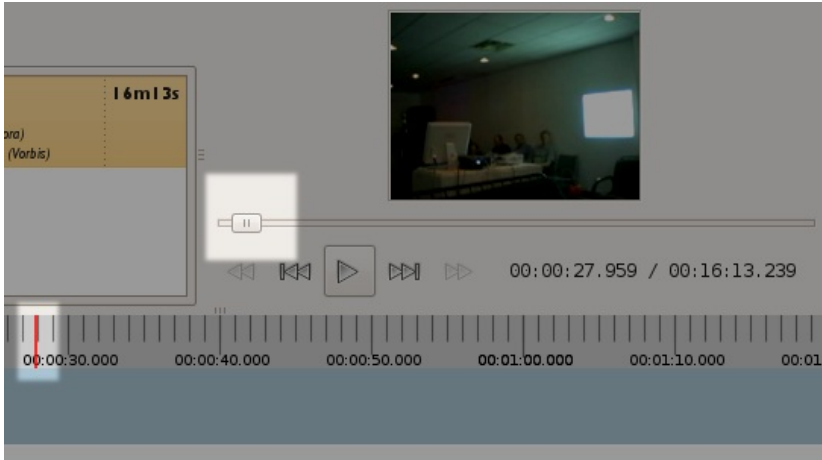


TOISTO

Kun videoleike on aikajanalla, et näe yksittäisiä kuvaruutuja, mutta voit vetää liikusäädintä eteenpäin ja taaksepäin nähdäksesi videon.



Kun toistat tällaista videota, voit nähdä toiston sijainnin (ajan) punaisella viivalla, joka on aikajanalla:

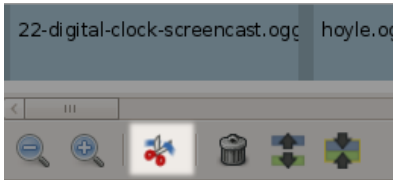


LIKKUVAT VIDEOLEIKKEET

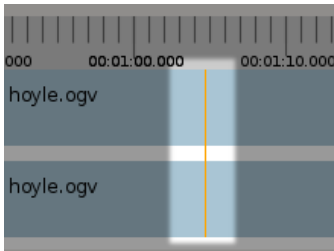
Voit siirtää videoleikkeitä ympäriinsä vetämällä niitä pitkin aikajanaa.

LEIKKEIDEN LEIKKAUS

Valitse saksi-ikoni leikataksesi leikkeen kahtia:



Siirrä sitten kursoria ympäri aikajanaa. Näet oranssin viivan ilmestyvän, se näyttää leikkauksen potentiaalisen sijainnin:



Kun oranssi viiva on oikeassa paikassa, klikkaa aikajanaa ja valittu videoleike leikataan kahtia siinä pisteessä.

VIDEOLEIKKEIDEN POISTAMINEN

Poistaaksesi videoleikkeen klikkaa sitä kursorilla, jotta se valitaan, ja paina sitten roskakori-ikonia:

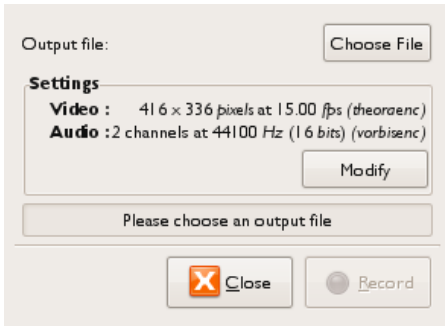


VIENTI OGG THEORAAN

Kun projektisi on valmis vietäväksi, voit klikata Piirrä Projekti -nappia:



Näet nyt renderöintiasetusten ikkunan:



Klikkaa Valitse tiedosto ja valitse paikka, johon tahdot tallentaa viedyn tiedoston. Voit myös muokata koodausasetuksia painamalla Muokkaa, kun asetukset sopivat tarpeisiisi, paina Tallenna ja hae kahvia.

ESIKATSELU

32. JOHDANTO

33. TEE ESIKATSELUKUVIA

34. TEE DIAesitys

35. TEE ESIKATSELUVIDEO

JOHDANTO

Jos tahdot leikkiä Ogg-tiedostoillasi, sinun ei tarvitse avata kokonaista editointiohjelmaa. Joukko hyödyllisiä ohjelmia on olemassa yksinkertaisten asioiden tekemiseen Ogg-tiedostojen kanssa komentoriviltä. **Ogg Video Tools** on työkalulaatikko Theora-videotiedostoja varten, se sisältää joukon käteviä komentorivin työkaluja Theora-tiedostojen manipulointia varten. Käytä niitä leikkaamiseen ja liittämiseen tai esikatselukuvien tekemiseen, ilman että täytyy keskittyä monimutkaiseen editointiprosessiin.

OGG VIDEO TOOLSIN ASENTAMINEN

Useimmat GNU/Linux -distrot tarjoavat Ogg Video Toolsin ohjelmavarastoissaan (esim. Fedora, openSuSE). Jos työkaluja ei ole saatavilla suosikkidistroosi, tai jos tahdot päivittää se uusimpaan versioon, jatka kohtaan asentaminen lähdekoodista alempana. Tämä pätee myös BSD ja Mac OS -käyttäjille.

Windowsin käyttäjät voivat ladata ZIP-tiedoston, joka sisältää ajettavat komentorivin työkalut, osoitteesta: <http://sourceforge.net/projects/oggvideotools/files/>.

Lähteestä asentaminen

Käy sivulla <http://sourceforge.net/projects/oggvideotools> ja klikkaa *Download Now* saadaksesi kopion lähdekoodista.

Ennen kuin jatkat asentamista, katso että kaikki vaadittavat kehityspaketit on asennettu. Ogg Video Tools riippuu seuraavista muista paketeista:

- theora ja theora-dev
- vorbis ja vorbis-dev
- ogg ja ogg-dev
- GD-lib ja sen kehitysosat
- SDL ja SDL-dev (jos tahdot kompiloida oggScrollin)

Ogg Video Tools kompiloituu onnistuneesti vaikka nämä kaikki kirjastot puuttuisivat, mutta silloin vain osa editointityökaluista kompiloidaan. Muut työkalut eivät toimi asennuksen jälkeen.

Kun olet varmistanut vaadittavien pakettien läsnäolon, avaa terminaali ja kirjoita seuraava komento (ilman alun dollarimerkkiä):

```
$ tar xzvf oggvideotools*.tar.gz
$ cd oggvideotools*
$ ./configure
$ make
$ su
<type root password>
$ make install
```

Jossa `./configure` -komennon kolmannella rivillä pitäisi antaa tämän tapainen ulostulo:

```
[ ... ]
checking for bzero... yes
checking for memset... yes
configure: creating ./config.status
```

Creating the following tools:

```
oggJoin
oggSplit
oggCut
oggCat
oggDump
```

```
oggSlideshow
oggThumb
```

oggResize
oggSilence
oggScroll

ESIKATSELUKUVIEN LUOMINEN

Paketissa Ogg Video Tools (<http://sourceforge.net/projects/oggvideotools>) on oggThumb -työkalu, joka tekee Theora-videotiedostoista esikatselukuvia.

Esimerkkikomento:

```
$ oggThumb -t1,2,3,4 -s320x240 my_theora_video.ogv
```

Tässä luomme neljä esikatselukuvaa videotiedostosta "my_theora_video.ogv", ne on otettu videon kestätyä 1, 2, 3 ja 4 sekuntia. Tässä esimerkissä esikatselukuvat tallennetaan seuraaviin tiedostonimiin:

```
my_theora_video_0.jpg  
my_theora_video_1.jpg  
my_theora_video_2.jpg  
my_theora_video_3.jpg
```

ESIKATSELUKUVIEN SARJAN LUOMINEN

Jos sinun täytyy tehdä esikatselukuvia määrätyn väliajoin koko videosta, tietämättä videon pituutta etukäteen, käytä seuraavaa shelliskriptiä homman hoitamiseen:

```
#!/bin/sh  
length=$(oggLength "$1")  
timestring="0"  
div=$((($2*1000))  
for((i=1; i<$2; i++)) do  
    value=$(echo "scale=3; $i*$length/$div" | bc -q 2>/dev/null)  
    timestring="$timestring,$value"  
done  
oggThumb "$3" -t$timestring "$1"
```

Skriptiä kutsutaan komennolla:

```
$ ./mkThumb my_theora_video.ogv 15
```

Tämä skriptikutsu ottaa 15 kuvaa videon alusta loppuun, yhtä pitkiltä etäisyyksiltä. Kuvat tulevat ulos tiedostoina, edellä kuvattujen nimeämiskäytäntöjen mukaan. Korvaa *my_theora_video.ogv* käyttämäsi videon nimellä. Esikatselukuvien määrä annetaan viimeisessä argumentissa. Muuta sitä sopimaan tarpeisiisi.

MÄÄRÄTYN KOKOISET ESIKATSELUKUVAT

Voit pyytää tietyn kokoisia esikatselukuvia lisäämällä option *-s* komentoriviin:

```
$ oggThumb -t1,2,3,4 -s320x240 my_theora_video.ogv
```

Tässä me luomme esikatselukuvia, joiden koko on 320x240. Voit asettaa korkeuden tai leveyden arvoksi 0, jolloin arvo lasketaan automaattisesti:

```
$ oggThumb -t1,2,3,4 -s120x0 my_theora_video.ogv
```

Tässä videon korkeus on asetettu vastaamaan leveyttä 120. oggThumb laskee korkeuden niin, että se säilyttää kuvan koon suhteen.

Kehittyneemmät toiminnot

Käytä aina optiota *-f* option *-t* sijasta valitaksesi esikatselukuvan aikasijainnin kuvaruudun numeron mukaan sekuntien määrän sijasta. Toinen maininnan arvoinen optio on *-o png*, joka luo esikatselukuvia PNG-kuvina, joilla on korkeampi laatu kuin oletusarvoisesti tehdyillä JPG-kuvilla.

oggThumb-komennon koko dokumentaatio löytyy kirjoittamalla:

\$ man oggThumb

komentorivillä. Huomaa, että Windows-versiossa on joukko HTML-tiedostoja, jotka ovat verkkoselaimessasi luettavassa käyttöoppaassa.

DIANÄYTÖKSEN LUOMINEN

Ogg Video Tools tarjoaa komentorivin työkalun, jota kutsutaan **oggSlideshow**ksi. Sillä voi luoda diaesityksiä kuvasarjasta JPG- tai PNG-muodossa ja käyttää erilaisia efektejä kuvien vaihtoa ja esitystä varten.

Rakenne: **oggSlideshow** [**Vaihtoehdot**] **kuva1 kuva2 kuva3**

Jokainen kuva luetaan tiedostojärjestelmästä ja sen kokoa muutetaan sopimaan videon ruutukokoon, jolle voidaan antaa -s -vaihtoehto. Jos kuvasuhde ei sovi videoruutuun, mustat reunat lisätään.

Seuraavia tehosteita tuetaan kuvien siirtymissä:

plain: Kuvat seuraavat toisiaan.

crossfade: Ensimmäinen kuva häipyä asteittain, seuraava ilmestyy aste asteelta. Ristihäivytyksen kesto on 1 sekunti.

blurring: Kuva epätarkennetaan ennen kuin se katoaa. Seuraava kuva näkyy epätarkkana, mutta tarkkenee vähitellen.

Eräs erikoistehoste on Ken Burns -efekti, joka on nimetty kuuluisan dokumentaristin mukaan. Tämä efekti lähentää kuvan läpi esityksen aikana. Tämän tehosteen siirtymä on asetettu ristihäivytykseksi.

On myös vaihtoehtoja bittinopeuden asettamiseksi, yhden kuvan esitysajan asettamiseksi, sekä ulostulotiedoston nimen määrittelemiseksi. Nämä on esitetty alla:

KOMENTORIVIN VAIHTOEHDOT:

-s

Asettaa videoruudun koon. Oletuskoko on 480x320.

Esimerkki: -s 320x240

-f

Asettaa videon ruutunopeuden, joka annetaan ruutuina sekunnissa. Oletusarvoinen ruutunopeus on 24 ruutua sekunnissa.

Esimerkki: -f 16

-o

Asettaa ulostulotiedoston nimen. Oletusarvoinen nimi on slideshow.ogv.

Esimerkki: -o myShow.ogv

-l

Asettaa kuvan esitysajan. Jos sinulla on 10 kuvaa ja asetat 10 sekunnin ajan, videon kokonaispituus on 1:40 (eli 100 sekuntia).

Esimerkki: -l 10

-d

Asettaa videon datanopeuden, joka mitataan bitteinä sekunnissa. Tämä on tarkoitettu ylemmäksi sietokynnykseksi. Lopullinen Theora-tiedosto voi olla odotettua pienempi.

Esimerkki: -d 1024000

-t

Asettaa kuvan siirtymän tehostetyypin, kuten yllä on kuvattu. Tuetut tyypit ovat:

- kb
Ken Burns -tehoste (oletus)
- cf
kuvan ristihäivytys siirtymään
- p
tavallinen kuvaesitys
- b
epätarkka siirtymä

Esimerkki: -t p

-e

Mahdollistaa ruudun koon muuttamisen. Jos kuva ei sovi videoruudun kuvasuhteeseen, se voidaan "uudelleenkehystää" lisäämällä mustat reunat. Tämä vaihtoehto on käyttökelpoinen vain Ken Burns -efektin kanssa (vaihtoehto -t kb). Muiden siirtymätehosteiden kanssa kuvat uudelleenkehystetään automaattisesti myös ilman -e vaihtoehtoa.

Esimerkki: -t kb -e

ÄÄNEN LISÄÄMINEN DIAESITYKSEEN

oggSlideshow voi tehdä vain mykkiä videotiedostoja. Äänen lisääminen jälkikäteen vaatii käyttämään vain muutamia muita työkaluja.

Olettakaamme, että loit diaesityksen käyttäen kaikkia kuvia hakemistostasi *joitainKuvia*, kuten tässä komentorivin esimerkissä näytetään:

```
$ oggSlideshow -s 320x240 -f 24 -o diaesitys.ogv -d 512000 joitainKuvia/*.jpg
```

Ennen kuin lisäät äänen, joudut ensin luomaan ääniraidan, jolla on sama pituus kuin diaesityksellä. Diaesityksen pituus tulostetaan diaesityksen luomisprosessin lopussa. Vaihtoehtoisesti sen voi nähdä käyttämällä seuraavaa komentoa:

```
$ oggLength diaesitys.ogv
```

Tämä palauttaa videon pituuden millisekunteinä. Tätä tietoa käyttäen voimme leikata Ogg/Vorbis äänistreamin saman pituiseksi. Tässä esimerkissä pituus on 5,6 sekuntia:

```
$ oggCut -l 5600 aaniRaitani.ogg muutettuAaniRaitani.ogg
```

Molempien streamien liittämiseksi yhteen mediatiedostoon (*newSlideshow.ogm*), käytämme komentoa `oggJoin`:

```
$ oggJoin uusiDiaesitys.ogv muokattuAaniRaitani.ogg diaesitys.ogv
```

Nyt olet valmis!

ALOITUSKUVAN LISÄÄMINEN

Diaesityksen ensimmäisen kuvan täytyy usein olla otsikkokuva, joka kertoo joitain tietoja diaesityksestä. Luo lyhyt ja yksinkertainen diaesitys, jota voit käyttää otsikkona, käyttäen vain yhtä kuvaa:

```
$ oggSlideshow -s 320x240 -f 24 -tp -o aloitusKuva.ogv -l 10 -d 512000 \ aloitusKuva.jpg
```

Tässä diaesityksen tyyppinä on *plain* (*-tp*), jolloin kuva näkyy sellaisena kuin se on. Kesto on 10 sekuntia (oletusarvo on 8 sekuntia).

Liitämme kuvan diaesityksemme *diaesitys.ogv* alkuun käyttäen komentoa *oggCat*:

```
$ oggCat kokoDiaEsitys.ogv aloitusKuva.ogv diaesitys.ogv
```

Tietenkin voimme myös lisätä ääniraidan, kuten aiemmin on kuvailtu:

```
oggJoin uusiDiaEsitys.ogv muokattuAaniRaitani.ogg kokoDiaEsitys.ogv
```

SKRIPTI KAIKKEA VARTEN

Seuraava pääteskripti yhdistää kaiken tässä luvussa esitellyn, se luo diaesityksen, jossa on alkutekstit ja äänet.

```
#!/bin/sh
#
# käyttö: ./diaesitys ~/kuvahakemistoni/ äänitiedosto.oga ulostulotiedosto.ogm
#
# Muutettavat muuttujat
#
# videon ruutukoko
SIZE="640x360"
#
# diaesitys-streamin datanopeus bitteinä sekunnissa
DATARATE="1024000"
#
# yhden kuvan esitysaika sekunteina
PR_TIME="10"
#
# ruutunopeus kuvina
FRAMERATE="24"
#
# uudelleenkehystä kuva
# Tämä lisää mustat kehykset kuvaan, jotta kuvasuhde pysyy ennallaan.
# Ken Burns -efektillä tämä ei ole välttämätönsä, mutta liukumata voivat olla
# tasaisempia.
#REFRAME="-e"
#REFRAME=""
#
# uudelleensämpläys
# Tämä vaihtoehto määrittelee, kuinka gdlib lataa kuvan.
# Koska gdlibin koonmuuttamismekanismi on todella hyvä, sitä käytetään
# tuomaan se arvoon, joka on lähellä videon ruutukokoa (yleensä vähän isompi).
# Et yleensä näe suurta eroa, mikäli muutat tätä arvoa. Pidä se ennallaan.
# (oletusarvo on 1.2).
RESAMPLE="1.2"
#
# diaesityksen tyyppi
# kb - Ken Burns -efekti (liukuu ja lähentää)
# p - tavallinen (kuva vain näkyy, ei ristiinhäivytystä kuvien välillä)
# cf - ristiinhäivytytys (kuvien näyttö, ristiinhäivytytys kuvien välillä)
TYPE="kb"
#
#
# Tilapäinen tiedostonimi
TMP_VIDEOFILE="slideshow_tmp.ogv"
TMP_AUDIOFILE="audio_tmp.oga"

# diaesityksen luominen
oggSlideshow -s $SIZE -d $DATARATE -l $PR_TIME -f $FRAMERATE \
  $REFRAME -r $RESAMPLE -t $TYPE -o $TMP_VIDEOFILE $1/*.jpg

# mikä tämän pituus on
LENGTHVIDEO=`oggLength $TMP_VIDEOFILE`

#
# leikkaa äänitiedostoa
```

```

LENGTHAUDIO=`oggLength $2`

#
# onko äänitiedosto liian lyhyt?
if [ $LENGTHVIDEO -gt $LENGTHAUDIO ]
then
    echo "varoitusta: äänitiedoston ($LENGTHVIDEO) on pidempi kuin äänitiedoston ($LENGTHAUDIO)"
    exit -1
fi

# äänitiedoston leikkaus
oggCut -l$LENGTHVIDEO -i$2 -o$TMP_AUDIOFILE

#
# Liitä ääni- ja videotiedosto
oggJoin $3 $TMP_VIDEOFILE $TMP_AUDIOFILE

#
# Poista vanhat tiedostot
rm -f $TMP_VIDEOFILE $TMP_AUDIOFILE

```

ESIKATSELUN LUOMINEN

Kun video laitetaan verkkoon on usein hyvä laittaa lyhyt esikatselu videosta pääverkkosivulle. Se voidaan linkittää alkuperäiseen videoon. Käytä komentorivin työkalua **oggResize** paketista Ogg Video Tools (<http://sourceforge.net/projects/oggvideotools>) luodaksesi sellaisen esikatselun.

Tyypillinen oggResize -komento näyttää seuraavalta:

```
$ oggResize -s256x144 -d64000 -f1 -p24 bigBuckBunny.ogv bigbuckbunnyPreview.ogv
```

Tässä luomme Theora -videotiedoston *bigbuckbunnyPreview.ogv* alkuperäisestä *bigBuckBunny.ogv* -tiedostosta ja skaalaamme sen pienen 256x144 pikselin kuvakokoon.

Optio -p saa oggResize-komennon sisältämään vain joka 24:n kuvan alkuperäisestä, lyhentää keston 1/24:n alkuperäisen pituudesta, luoden yleensä nopeasti eteenpäin kelaavan version videosta. Koska ei ole paljonkaan järkeä kelaata audiota nopeasti eteenpäin, äänet on poistettu.

Tässä olemme myös lisänneet option -f1, mikä säätää luodun videon kuvataajuutta yhteen kuvaruutuun sekunnissa. Tämä estää videon nopean kelaamisen eteenpäin, sitä on helpompi katsoa, ja se näyttää samalta kuin animoitu GIF-kuva.

LISÄÄ SOITTONAPPULA

FE ensin tarvitsemme PNG-kuvan nappulasta, jossa on läpinäkyvä maski, jonka avulla se voidaan laittaa videosisällön päälle. Läpinäkyvää maskia sanotaan alfakanavaksi, nykyisin useimmat kuvankäsittelyohjelmat antavat luoda yhden. Kokeile esimerkiksi vapaata kuvaeditoria GIMP.

Optiot -A ja -a antavat sinun laittaa kuvan ulostulovideon päälle ennen kuin video skaalataan uudestaan vastaavasti kokoon, jonka -s antaa.

```
$ oggResize -s256x144 -d64000 -f1 -p24 -A Play-256x144.png bigBuckBunny.ogv \
bigbuckbunnyPreview.ogv
```

Siinäpä se! Nyt sinulla on hieno esikatselu videosta *bigBuckBunny.ogv*. Sen päällä oleva soittonappi kutsuu ihmisiä klikkailemaan itseään.

EDITOINTI KOMENTORIVILTÄ

36. JOHDANTO

37. THEORAN TEORIA

38. TIEDOSTON JAKAMINEN

39. TIEDOSTON LEIKKAAMINEN

40. TIEDOSTOJEN LIITTÄMINEN

41. TIEDOSTOJEN YHDISTÄMINEN

42. KOON MUUTTAMINEN

43. OGGIN ANALYSOIMINEN

JOHDANTO

Komentorivi on tekstiin perustuva menetelmä hyötyohjelmien kutsumiseen päätteen komentotulkit. Useimmissa Linux-käyttöjärjestelmissä on ohjelma, jota kutsutaan päätteeksi. Useimmiten tämä pääte on Bourne-Again Shell, "bash", joka on alkuperäisen Bourne Shellin, "sh", seuraaja, mutta monia erilaisia päätteitä on käytössä.

Jokaisen käyttäjän tulisi tietää, kuinka voi käyttää yksinkertaisia tekstikomentoja, kuten "df -h", joka on levytilan käyttöraportti ihmiselle ymmärrettävissä yksiköissä (megatavuina levyn sektoreiden sijaan). Komentorivin perusteet ja monia edistyneempiä asioita on dokumentoitu FLOSS Manualsin kirjassa *Komentorivin perusteet*, joka on saatavilla osoitteessa <http://fi.flossmanuals.net/KomentorivinPerusteet/Introduction>. Tuo kirja on hyvä lukea ennen kuin kokeilee mitään tässä luvussa kuvailtuja asioita. Kirjan avulla on helppo ymmärtää komentorivin perusteita, jolloin on mahdollista astua kehittyneemmän käytön kynnyksen yli. Kehittyneempi komentorivin käyttö Linuxissa merkitsee kahta asiaa: kaikkien komentojen sisältyvien vaihtoehtojen käyttöä (jotka man-komento listaa - kokeile komentoa "man man"), ja muuttamaan riviä pidempien ohjelmien kirjoittamista käyttäen muitakin kuin yksinkertaisimpia komentoja.

Monet videoeditointityökalut toimivat sekä koko ruudun graafisena käyttöliittymänä että komentoriviversiona. Graafinen käyttöliittymä kuitenkin käyttää usein komentoriviä tehtävien suorittamiseen. Esimerkkinä tästä ovat Ubuntu ja Debianin käyttämät pakettinhallintaohjelmat. Koko ruudun graafinen käyttöliittymä on Synaptic, ja tekstiin perustuva käyttöliittymä on Aptitude. Hyvin kirjoitettu käyttöliittymä voi tehdä komentojen käytöstä paljon helpompaa kuin komentorivin käytöstä, mutta komentorivin käyttö antaa paljon suuremmat mahdollisuudet komentojen kehittyneempään käyttöön. Lisäksi komentorivin komennot voi tallentaa tekstitiedostoihin, jolloin niistä tulee ajettavia ohjelmia, kunhan tekstitiedostoista tekee suoritettavia chmod-komennon avulla.

TAPPIOTON VIDEOEDITOINTI

Videoeditointi on hyvin kallis hanke tietokoneesi laskentatehon ja muistinkäytön suhteen. Tämä johtuu siitä, että monimutkaiset videoeditointityökalut joutuvat antamaan sinulle mahdollisuuden työskennellä videon jokaisen ruudun ja äänen jokaisen nämpelen parissa.

Jokaisen yksittäisen ruudun käsittely ei kuitenkaan ole aina tarpeen. Joissain tapauksissa ihmiset tahtovat vain erottaa ääni- tai kuvavirran tiedostosta, tai yhdistää ääni- ja kuvavirrat yhteen streamiin, tai he eivät välitä videon leikkauksesta yhden ruudun tarkkuudella.

Tällöin video- ja äänivirtoja ei tarvitse koodata uudelleen, mikä tekee videon käsittelystä paljon vähemmän tietokoneen resursseja vaativaa. Se merkitsee myös, että editointityökalut ovat hyvin nopeita.

Tätä editointiprosessia kutsutaan tappiottomaksi, sillä uudelleenkoodausta ei vaadita. Videon uudelleenkoodaus aiheuttaa aina videon laadun heikkenemisen ja videon editointi ilman uudelleenkoodausta on täten tappiotonta. Koodatut paketit, jotka ovat joko videota tai ääntä, eivät joudu toimenpiteiden kohteiksi. Ne pidetään sellaisina kuin ne ovat, joten ne koodataan samalla laadulla, joka niillä oli ennen editointiprosessia.

TYÖKALUT

On kaksi pääasiallista työkalupakkia tappiottomaan videoeditointiin: **Ogg Video Tools** ja **Oggz Tools**. Molemmat ovat komentorivityökalujen kokoelmia. Ogg Video Tools ja Oggz Tools toimivat Linuxissa, OSX:ssä ja Windowsissa.

Oggz löytyy täältä: <http://www.xiph.org/oggz/>

Ogg Tools löytyy täältä: <http://sourceforge.net/projects/oggvideotools>

Joissain tapauksissa nämä kaksi työkalupakkia omaavat samanlaisen toiminnallisuuden. Oggz Tools on suuntautunut kehittäjiä varten ja Ogg Video Tools on suuntautunut loppukäyttäjiä varten. Ogg Video Tools tarjoaa työkaluja ogg-videotiedostojen luomista ja käsittelyä varten, mutta Oggz Tools tarjoaa joukon muita työkaluja, varsinkin ogg-tiedostojen analysointia varten.

Ogg Video Tools voi käsitellä videostreameja, jotka on luotu Theora-koodaajalla ja äänistreameja, jotka on luotu Vorbis-koodaajalla. Oggz voi käsitellä muita stream-formaatteja, kuten FLAC, Speex, CELT jne.

Ogg Video Tools komentorivityökaluihin sisältyy:

- **oggSplit:** jakaa useamman streamin ogg-tiedoston useampaan tiedostoon, joista jokaisessa on yksi streami.
- **oggJoin:** yhdistää Ogg-ääni (vorbis) ja Ogg-video (theora) -tiedostot yhteen Ogg-tiedostoon.
- **oggCut:** luo uuden Ogg-tiedoston alkuperäisen Ogg-tiedoston aliosana.
- **oggCat:** liittää kaksi tai useampia Ogg-tiedostoja.
- **oggResize:** muuttaa Theora tai Vorbis -streameja useammilla tavoilla.
- **oggDump:** tulostaa kasan paketteja tai sivuja annetun Ogg-tiedoston sisältä.
- **oggLength:** palauttaa annetun Ogg-tiedoston pituuden.
- **oggSilence:** luo vorbis-äänitiedoston, jolla on annettu pituus, ja joka on täynnä hiljaisuutta.

Oggz Tools -komentorivityökalut sisältävät:

- **oggz-chop:** erottaa osan Ogg-tiedostosta annettujen alku- ja loppuaikojen välistä.
- **oggz-merge:** yhdistää Ogg-tiedostot yhteen, lomittaen tiedostot esitysajajärjestykseen.
- **oggz-sort:** järjestää Ogg-tiedoston sivut esitysajan mukaan.
- **oggz-rip:** erottaa yhden tai useamman loogisen bittistreamin Ogg-tiedostosta.
- **oggz-info:** näyttää tiedon Ogg-tiedoston käyttämisestä koodekeista.
- **oggz-comment:** listaa tai muokkaa kommentteja Ogg-streamin otsikkojen sisällä.
- **oggz-validate:** tarkastaa Ogg-tiedoston yleisten ongelmien varalta.
- **oggz-dump:** tulostaa vedoksen pakettistreameista annetun Ogg-tiedoston sisällä.

VÄHÄN TEORIAA

Theora-videon editointi käyttäen komentorivin työkaluja, kuten Oggz Tools tai Ogg Video Tools, ei vaadi tietoa Theoran toiminnan tarkoista yksityiskohdista. Kuitenkin nämä komentorivin työkalut prosessoivat Theoran tiedostoja hyvin matalalla tasolla. Joitain sivuvaikutuksia ilmenee, näitä sivuvaikutuksia ei voida selittää ilman Theora-formaatin yksityiskohtien käsittelyä.

Jos uskot, että voit elää pienten editoinnin aiheuttamien epätarkkuuksien kanssa, voit hyvin jättää tämän luvun väliin.

THEORA-VIDEON ANATOMIA

Videotiedosto sisältää normaalisti videostriimin ja audiostreamin. Molempien striimien varastointi yhteen tiedostoon vaatii säilytysformaattia (container format). Säilytysformaatti Theora-videota varten on Ogg, joka pitää sisällään Theora-videostriimin ja yhden tai useampia Vorbis-audiostreameja. Audio ja videostriimit tallennetaan välikkain. Tämä merkitsee, että jokainen striimi on segmentoitu muutamaan datablokkiin, jotka ovat lähes samaa kokoa. Näitä blokkeja kutsutaan sivuiksi (page).

Jokaisessa sivussa on aikaleima, joka antaa informaatiota sivun sijoituksesta striimin sisällä.

Video- ja audiostreamit ovat toistensa lomassa siten, että molempien striimien sivut on liitetty yhteen nousevassa järjestyksessä aikaleimojensa mukaan.

DEMUPLEKSAUS

Teknisesti on helppoa jakaa videotiedosto video- ja audiostreameihin. Tämä johtuu siitä seikasta, että kaksi striimiä voidaan erottaa keräämällä video- ja audiosivut erillisiin tiedostoihin. Videotiedoston striimeihin jakavaa prosessia kutsutaan demultipleksaukseksi.

Kaikki tarpeellinen toiston vaatima tieto on kerätty striimien sisään, ilman että Ogg-säiliö lisää mitään tietoa. Jokainen tiedostoista itsessään on standardeihin sopiva Ogg-säilytys striimi ja sitä voi lukea mikä tahansa Theora/Vorbis-yhteensopiva video- tai audiosoitin.

SIVUN AIKALEIMAT

Kuten aiemmin on mainittu, striimit tallennetaan jaettuna useiksi sivuiksi. Sivulla on otsikko, joka pitää metadatat, joka kuvaa tämän osan video/audiostreamista. Otsikkoon sisältyy ajastusinformaatio, ainutlaatuinen striimintunnistusnumero, sivun numero ja jotain muuta tietoa.

Ogg-standardien mukaan ajastustieto jokaiselle sivulle annetaan jyväsijainnilla, 64-bittisellä arvolla, joka on otsikossa. Kuinka jyväsijainti liittyy yhteen todellisen aikasijainnin kanssa on striimin määrittelemää, se voi olla esimerkiksi määrä millisekunteja videon alusta. Tästä syystä Ogg-tiedostoja käsittelevän ohjelman pitää sisältää jyväsijainnin tulkitsija, jotta se voi käsitellä tiedostoa oikein. Tämän tulkitsijan pitää olla tietoinen koodeksista ja striimispesifistä informaatiosta, jotta se voi tuottaa ajastustietoa, jota voidaan vertailla eri striimien välillä.

Ogg-standardi tarkoittaa, että kaikki sivut Ogg-tiedostossa täytyy tallentaa nousevassa aikajärjestyksessä, vaikka se ei määrittele jyväsijaintien tulkintaa. Joten mikä tahansa työkalu, joka leikkaa tai yhdistää striimeja tarvitsee toimivia jyväsijaintitulkkeja jokaista sisällettyä striimiä varten, jotta sivut voidaan lomitaa oikein.

KOODEKKIDATAN KOTELOIMINEN

Sivut ovat miltei samaa kokoa oletusasetuksena (noin 4096 tavua). Kuitenkin audio- ja videopakettit, jotka tietty koodekki luo, eivät yleensä sovi täsmälleen sivulle. Audiopakettit ovat yleensä paljon pienempiä. Videopakettit voivat olla monen kokoisia, pienempiä tai suurempia, riippuen eri tekijöistä.

Video- ja audiokoodekkien tuottama data koteloidaan ensin Ogg-paketiksi. nämä paketit sijoitetaan sitten Ogg-sivuille. Paketti voidaan tarpeen mukaan joko jakaa useammalle sivulle, tai yhdistää muiden pakettien kanssa, jotta muuten osittain tyhjä paketti täyttyy.

Theoran videodatan koteloiminen

Koodaajan kehittämä Theoran videodata sisältää kahdenlaisia Ogg-paketteja: niin sanottuja avainkuvaruutuja (kutsutaan usein I-Frameksi), jotka ovat kokonaisia kuvia, ja P-Frameja, jotka kantavat ainoastaan viimeisimmän ja nykyisen kuvan eroja.

Jotta annettu kuvaruutu voidaan näyttää annettuna aikana, dekooderin täytyy tietää edellinen avainkuvaruutu ja sen täytyy dekodata kaikki kuvaruudut (mukaanlukien avainkuvaruutu) annetun kuvaruudun tosiasialliseen aikasijaintiin asti.

Jotta videota voi dekodata, dekooderi tarvitsee tietoa striimistä itsestään, esimerkiksi videon kuvaruutujen koosta. Tämä informaatio on sijoitettu otsikkopaketteihin striimin alkuun.

Vorbis-audiodatan koteloiminen

Enkooderin luoma Vorbis-audiodata kantaa tietyn määrän audionäytteitä. Näyte on audiodatan yksikkö. Näytteiden määrä Ogg-paketissa on määrätty ja se voi vaihdella vain kahden mahdollisen koon välillä, kuten striimin otsikossa on määritelty.

Videodatan tapaan audiodatapaketit ovat myös riippuvaisia toisistaan. Jotta yksi audiopaketti voidaan dekodata, edellistä pakettia tarvitaan.

Audion dekoodaaja riippuu striimin parametreista, kuten näytetaajuudesta, bittitaajuudesta, jne. nämä on tallennettu striimin otsikon paketteihin striimin alussa.

OGG-KEHIKKO

Kuten aiemmin sanottiin, videostriimi ei voi alkaa satunnaisesta ajankohdasta, sillä siinä on avainkuvaruutuja (I-Frame) ja delta-kuvaruutuja (P-Frame). Lisäksi audio on tallennettu paketteihin, joille on annettu ajoitusjyvät.

Jotta synkronisaatio ja oikea aloituskohta on varmistettu, Ogg-kehikkosivut kantavat tietoa jokaisen striimin aloituspaikasta. Dekoodaaja, joka lukee Ogg-kehikon tietoa, voi sitten etsiä oikean audio- ja videokohdan ja alkaa soittamaan tästä kohdasta.

TIEDOSTON JAKAMINEN

Videotiedoston jakamista sen osastriimeihin kutsutaan usein de-multipleksaukseksi. Komentorivin työkalu `oggSplit` Ogg Video Tools -paketista tekee demultipleksauksen.

Komento: **oggSplit alkuperäinenTiedosto.ogv**

Tämä komento uuttaa tiedoston *alkuperäinenTiedosto.ogv* eri striimit erillisiin tiedostoihin. Se käyttää seuraavaa nimeämisjärjestelmää luoduille tiedostoille:

<koodekinNimi>_<ID>.<pääte>

KoodekinNimi on *theora*, *vorbis* tai *unknown*, riippuen striimin koodekin tyypistä.

Jokaisella striimillä on erillinen *tunnusnumero (ID)*, jonka asettaa koodaaja. Tämä ID on 32-bittinen numero, joka on yksilöllinen joka striimissä yhden Ogg-tiedoston sisällä. Koska voi olla enemmän kuin yksi ääni- tai videostriimi tiedoston sisällä (eli useita eri audiostreameja eri kielille), tätä ID:tä tarvitaan erottamaan ne toisistaan.

Pääte asetetaan sopimaan striimin sisältöön. *ogv* on käytössä videotiedostoille, *oga* äänitiedostoille ja tuntemattomat striimit saavat päätteen *ogg*.

Esimerkki:

```
# oggSplit myFile.ogv
# ls
theora_a1bb23c1.ogv      vorbis_a316522.oga
```

OGG-TIEDOSTON LEIKKAUS

Video- ja audiostriimeistä muodostuvan Ogg-tiedoston leikkaaminen osiin ajan mukaan, säilyttäen samalla synkroniteetin ulostulon tiedostojen striimien välillä, on hieman hankalaa. Tämän vuoksi sen tekemiseen on monta eri tapaa, tämä luku antaa sinulle kuvan mahdollisuuksista ja niiden hyvistä ja huonoista puolista.

Videotiedostojen editointiin käytetyt työkalut voidaan jakaa tappiottomiin ja tappiollisiin työkaluihin. Tappiottomat työkalut käsittelevät vain Ogg-striimin (ogg-sivut ja ogg-paketit). Niinpä datan laatu säilyy. Tappiollisten työkalujen täytyy ensin dekodata data mahdollistaakseen editoinnin ja sitten koodata ne uudestaan, mikä johtaa laadun menetykseen. Tällainen editointi tarjoaa suurempaa joustavuutta (esimerkiksi se sallii kuvien asettamisen toistensa päälle) laadun kustannuksella. Tämän lisäksi dekodaus ja koodaus -prosessit vaativat paljon enemmän aikaa kuin pelkkä Ogg-striimin pakettien käsittely.

Tämä kappale käsittelee vain työkaluja, jotka leikkaavat videotiedostoja tappiotta.

KOMENTO OGGCUT

Komentorivin työkalu oggCut on työkalu, joka voi leikata Ogg-tiedostoja kuva kuvalta. Käytä sitä näin:

```
oggCut -s <StartTime> -e <EndTime> originalFile.ogv createdFile.ogv
```

jossa <StartTime> ja <EndTime> annetaan millisekunteina. Vaihtoehtoisesti voit leikata käyttäen aloitusaikaa ja pituutta optiolla -s:

```
oggCut -s <StartTime> -l <Length> originalFile.ogv createdFile.ogv
```

Myös tässä <Length> mitataan millisekunteina.

Kun oggCut ajetaan, se etsii ensimmäistä Ogg Theora -pakettia, jossa aikaleima on myöhemmin tai samaan aikaan -s optiolla annetun ajan kanssa. Jos tämä paketti löytyy, ohjelma etsii seuraavan avainkuvaruuden ja luo uuden tiedoston, jossa tuo avainruutu on ensimmäinen ruutu.

Kun ohjelma löytää paketin, jossa aikaleima on suurempi kuin loppuaika, se sulkee uuden tiedoston ja lopettaa.

Aloitusaika ei luultavasti ole avainkuvaruuden sijainti, luodusta tiedostosta tulee luultavasti oletettua pienempi.

Videon ja audion synkronisointiongelmat oggCutilla

Koska oggCut toimii vain pakettien rajoilla, ja koska Vorbis ja Theora striimipaketit eivät ole synkronisia, audio- ja videopakettit alkavat yleensä eri aikaan.

Kun oggCut luo uuden tiedoston, joka alkaa aikaan t , seuraava audiopaketti tulee luultavasti alkamaan hetkellä $t+x$, jossa x on aikaväli avainruudun videon ja seuraavan audiopakettin välillä.

Kun nämä kaksi striimiä kirjoitetaan uuteen tiedostoon, aikaväliin liittyvä informaatio hukataan. Niinpä soittimen täytyy olettaa, että molemmat tiedostot alkavat ajasta 0. Niinpä audiostriimi on aina hieman etujassa. Kuitenkin, koska audioblokki on aika lyhyt, ero on yleensä vain muutama millisekunti ja sitä ei juurikaan huomaa.

KOMENTO OGGZ-CHOP

Ohjelma oggz-chop paketista oggz-tools leikkaa Theora-videoita mistä tahansa pyydetystä kohdasta. Jos välttämätöntä, se leikkaa ei-avainkuvaruutujen kohdalta tai jopa keskeltä ruudun esitysaikaa.

Teknisesti tämän kaltainen hienojyväinen leikkaus ei ole mahdollista Theora-videotekniikalla. Se saavutetaan tempulla: sen sijaan että video todella leikattaisiin pyydetystä kohdasta, se leikataan lähimmästä sijainnista, jolloin tuloksena oleva videotiedosto on pidempi kuin pyydetään. Sitten niinsanottu Ogg-kehiksen otsikko lisätään videoon, ja se ohjaa soittimen soittamaan vain todellisuudessa pyydetyn osan videosta.

Ylimääräiset videoruudut, joita ei voitu leikata videosta, ovat edelleenkin paikallaan, ne on vain piilotettu toiston aikana.

Tämän toimintatavan huonot puolet ovat selkeitä: toisaalta levytilaa hukataan tallentamalla turhaa tietoa, toisaalta leikattu uusi tiedosto toistuu oikein vain soittimissa, jotka ymmärtävät Ogg-kehikon tiedoja. Toinen vähemmän selkeä ongelma on, että tällä tavalla tehtyjä tiedostoja ei voi liittää ilman, että piilotettu osa ilmestyy esiin. Tämä johtuu siitä, että Ogg-kehys voi piilottaa vain kuvaruutuja videon alussa ja lopussa. Kahden videon yhdistäminen tällä tavalla jättää kätkeytyjä ruutuja videon keskelle - tätä Ogg-kehys ei tue.

Kun nyt ymmärrät oggz-chopin käyttöön liittyvät ilmiöt, voimme kokeilla sen käyttöä:

```
oggz-chop -s <StartTime> -e <EndTime> originalFile.ogv createdFile.ogv
```

Tässä <StartTime> ja <EndTime> on spesifioitu muodossa <Hour>:<Minute>:<Second>. <Second> voi sisältää murto-osan, joten olet vapaa spesifioimaan sen millä tahansa resoluutiolla.

Meidän täytyy mainita toinen vaihtoehto: voit ajaa oggz-chopin lipulla -k pakottaaksesi sen jättämään pois kaiken Ogg-kehiksen tiedon. Tällä tavalla leikkauksen jyvällisyys on rajoitettu avainkuvaruutuihin ja voit olla varma, että luotu tiedosto toistuu oikein jopa vanhemmissa soittimissa, jotka eivät tue kehystä.

STREAMIEN LIITTÄMINEN

Ogg Video Tools tarjoaa komentorivin työkalun **oggJoin** kahden tai useamman streamin liittämiseksi yhteen Theora-videotiedostoon. Se voi tulkita sivun jyvän sijainnin ja liittää streamit, mikäli ne sisältävät Theora-videota tai Vorbis-ääntä.

Tätä prosessia kutsutaan myös multipleksaukseksi.

Synopsi: **oggJoin uusiTiedosto.ogv streami1.ogv streami2.oga [streami3.oga [...]
]**

Huomaa: Videostreamin (.ogv) täytyy olla ensimmäisenä.

Vaikka oggJoin sallii streamien yhdistämisen mihin tahansa järjestykseen, Theora-standardi vaatii, että videostreami on ensimmäinen streami. Esitettynä oikein: Theoran otsikon täytyy olla ensimmäinen otsikko multipleksatussa tiedostossa, jotta soittimet voivat erottaa pelkästään ääntä sisältävät tiedostot videotiedostoista. Tämä saavutetaan asettamalla videostreami ensimmäiseksi streamiksi listalla, kuten synopsissa näytetään.

OGG-TIEDOSTOJEN LIITOSTAMINEN

Liittääksesi kaksi tai useampia Ogg-tiedostoja, joilla on audio- ja videostriimit, yhteen videotiedostoon ilman uutta koodausta on helppoa, mutta se voi johtaa samoihin synkronisointiongelmiin kuin oggCutin käyttö.

Komentorivin työkalu oggCat luo uuden Ogg-tiedoston joukosta muita Ogg-tiedostoja.

Komento:

```
oggCat outputFile.ogv inputFile1.ogv inputFile2.ogv [ inputFile3.ogv [ .. ] ]
```

Kaikkien striimien parametrien täytyy sopia yhteen, jotta voit luoda Ogg-tiedoston. Jos yksi tai useampi syöttötiedoston parametri ei sovi yhteen, se tiedosto jätetään huomiotta, mutta ulostulotiedoston luomisen prosessi jatkuu. Ulostulotiedosto on toimiva Ogg-tiedosto. Se ei vain ole se Ogg-tiedosto, jota haluttiin.

PROSESSI

oggCat lukee ensin tiedoston niin kuin se on. Kuitenkaan se ei käsittele tuntemattomia striimejä eikä integroi niitä ulostulotiedostoon.

Löytääkseen vastaavan striimin seuraavista Ogg-tiedostoista, oggCat luo parametrityökalun, jossa on kaikki tarpeellinen tieto ensimmäisen tiedoston striimeistä (inputfile1.ogv), jolloin ensimmäinen tiedosto toimii "referenssinä".

Seuraavissa tiedostoissa (inputfile2.ogv inputFile3.ogv ...) olevat striimit on testattu viitetaulukon parametrejä vastaan, joten ne eivät voi olla samassa järjestyksessä viitetiedoston kanssa.

Esimerkki:

Ensimmäisessä Ogg-tiedostossa on yksi Theora ja yksi Vorbis -striimi. Theora-striimin videolla on 320x240 pikselin kuvakoko ja Vorbis-audiostriimillä on 32 kHz näytetaajuus.

Seuraava tiedosto sisältää kolme striimiä: kaksi Theora-striimiä ja yhden Vorbis-striimin. oggCat-komento voi yhdistää ne niin kauan kuin on ainakin yksi Theora-striimi, jossa on yhteensopivat parametrit (320x240 pikselin kuvakoko) ja audiostriimin parametrit sopivat myös yhteen. Toisessa Theora-striimissä voi olla mitä tahansa parametrejä.

Jos yhdenkään striimin parametrit eivät ole yhteensopivat, oggCat tulostaa rivin, joka ehdottaa kuinka tiedosto voitaisiin koodata uudestaan:

```
vorbis parameter compare: data rate not matching 32000 != 64000  
You may try to reencode with the datarate of the other file
```

```
Please try to resample with the following command  
oggResize -D 32000 <file see below>
```

CAT-KOMENNON KÄYTTÖ

Toinen tapa yhdistää kaksi videota on yhden tiedoston liittäminen toisen tiedoston perään komentorivityökalulla **cat**. Tämä metodi laukaisee soittimen aloittamaan uuden dekodaussektion seuraavaa striimiä varten.

Ogg-standardi sallii striimin avaamisen toisen striimin loppumisen jälkeen, mutta useimmat soittimet eivät toteuta tätä dekooderin session avauksen menetelmää. Niinpä metodi tulisi aina testata soittimilla, joita voidaan käyttää tämän session jälkeen.

OGGCAT JA OGGZ-CHOP

Kun tiedosto on luotu oggz-chop -komennon avulla, yhdistäminen ei ole mahdollista, sillä oggCat toimii eri tavalla.

Tämä johtuu siitä, että tiedostot, joissa on kehikko, ja eri aikamittaukset, jotka on määritelty tässä kehikossa joka striimille (kuten on tapaus useimpien oggz-chopilla luotujen tiedostojen suhteen) ei voida liittää yhteen jatkuvaan striimiin. Tämän käytöksen aiheuttaa se tosiseikka, että kaksi ogg-striimiä voisi mennä päällekkäin. Esitetyt kuvaruudut olisivat esittämättömien päällä, koska niitä tarvitaan vain ensimmäisen videoruudun dekodaukseen. Tämä ei ole mahdollista yhdessä Ogg-striimissä.

OGGCATIN SYNKRONISOINTIONGELMAT

Yhden Ogg-tiedoston eri striimit eivät välttämättä ole (täsmälleen) samanpituisia.

Yhdistämisprosessin aikana seuraava Ogg-tiedosto yhdistämislistalla laitetaan suoraan vastaavien edellisen tiedoston striimien perään ilman mitään aikaväliä. Niinpä seuraavan tiedoston synkronisointi kärsii ensimmäisen tiedoston epätarkasta striimin päättymisestä.

OGGRESIZE

Komentorivin työkalu **oggResize** voi muuttaa ogg-tiedoston (ogg, oga tai ogv) leveyttä ja korkeutta eri tavoilla. Kuitenkin se käsittelee vain Ogg Theora ja Vorbis -striimejä. oggResizen etu muihin työkaluihin verrattuna on, että se muokkaa vain striimit, joita täytyy muuttaa. Striimit, jotka eivät tarvitse muunnosta, jäävät vaille vaikutuksia.

oggResize voi muuttaa **videoruudun kokoa**, videon tai audion **datataajuutta** tai videon **kuvataajuutta** tai äänen **näytetaajuutta**.

Se voi myös lisätä **kommentteja** mihin tahansa striimiin ja voi sisältää **PNG-kuvia**, joissa on alfa-kanava, joten ne renderöidään videoon milloin tahansa ennen tai jälkeen koon muuttamisen.

Komento:

```
oggResize [optiot] inputfile.ogv outputfile.ogv
```

Mahdolliset optiot ovat:

-s *i* asettaa videoruudun koon. Koko annetaan muodossa *<levets>x<korkeus>*. Oletuskoko on 480x320.

Oletuksena videoruudun suhteet pidetään samoina, joten jos uuden videoruudun koko ei sovi suhteisiin, oggResize lisää mustat rajat ylös, alas, oikealle ja vasemmalle, jotta se sopii uusiin kokosuhteisiin.

jos tahdot venyttää videoruudut uuteen kokoon, käytä **-t** -optiota.

Esimerkki: -s 320x240

-d asettaa bittinopeuden bitteinä sekunneissa videokoodaajaa (theora) varten. Tämän on tarkoitus olla ylempi sietokynnys. Tiedosto voi siis olla oletettua pienempi. Jos sitä ei ole asetettu, alkuperäisen striimin bittinopeutta käytetään.

Esimerkki: -d 1024000

-D asettaa bittinopeuden bitteinä sekunnissa audiokoodaajaa (vorbis) varten. Jos sitä ei ole asetettu, alkuperäisen striimin bittinopeutta käytetään.

Esimerkki: -D 64000

-f asettaa videon ruutunopeuden numeraattorilla ja denominaattorilla, ja on ruutuja sekunnissa. Jos vain yhtä numeroa käytetään, denominaattori on 1. Jos tätä ei aseteta, alkuperäisen videon ruutunopeutta käytetään.

Esimerkki: -f 25:2

-F asettaa audiodatan näytetaajuuden Hertseinä. Jos näytetaajuus ei sovi alkuperäiseen tiedostoon, uudelleensämpläystä käytetään.

Esimerkki: -F 32000

-t jos tämä optio on asetettu, kuvaa venytetään ulostulokokoon. Tässä tapauksessa ulos tulevan videon kuvasuhde ei välttämättä vastaa alkuperäisen videon ruudun kuvasuhdetta. Tämä optio auttaa poistamaan mustat reunat, jotka ilmestyvät, jos alkuperäisen videon ja uuden videon kuvasuhteet eivät sovi yhteen.

Esimerkki: -t

Esimerkki: -c 'AUTHOR=yorn;DATE=03.07.09'

-C lisää kommentteja audion (vorbis) striimiin. Kommentit ovat muodossa 'tyyppi=arvo'. Useampi kuin yksi kommentti voidaan yhdistää puolipisteellä. On suositeltavaa, että heittomerkkejä (') käytetään, sillä komentorivi voi pitää puolipistettä välimerkinä.

Esimerkki: -C 'AUTHOR=yorn;DATE=03.07.09'

-q asettaa laadun. Arvot voivat olla väliltä 1 (paras laatu, hieman epätarkkuutta) ja 6 (huonoin laatu). Oletusarvo on 2.

Esimerkki: -q1

-p luo esikatselufilmin. Tälle optiolle annettu numero valitsee väliajan valittujen ruutujen välillä, esimerkiksi -p24 merkitsee, että joka 24:s ruutu on valittu, näytetty. Uudelleen luotu video toistuu siis 24 kertaa nopeammin. Tämä optio voidaan yhdistää -f -option, joka kontrolloi ruutunopeutta. Molemmilla optiolla voidaan luoda miellyttäviä videoesikatseluita. Jos -p on käytössä, audiostriimi jätetään huomiotta.

Esimerkki: -p 24

-a lisää kuvan videoruutuun ennen sen koon muuttamista. Kuvan ei tarvitse sopia videoruutuun. Se sijoitetaan aina videoruudun vasempaan yläkulmaan.

Esimerkki: <picture1.png>[,<startTime>[,<endTime>[,s]]]

Oletusarvoinen aloitusaika on 0. Oletusarvoinen lopetusaika on -1. Oletusarvoisesti s ei aseteta. Tässä tapauksessa kuva ei häivy pimeyteen. Jos s on asetettu, kuva häipy pimeyteen.

On mahdollista asettaa enemmän kuin yksi kuva videoruutuun. Käytä kaksoispistettä (:) yhdistääksesi ilmaukset. Jos aikajanaat ovat päällekkäin, kuvat asetetaan toistensa päälle, jolloin viimeinen kuva on etualan taso.

Esimerkki: -a etwas.png,2,7,s:etwasneues.png,5,10

-A lisää kuvan videoruutuun sen koon muuttamisen jälkeen.

Syntaksi on sama kuin ilmaisussa -a -option kanssa.

ESIMERKKEJÄ

Videoruudun koon ja bittinopeuden muutos:

oggResize -s320x240 -d512000 orig.ogv new.ogv

Tämä komento muuttaa videotiedoston orig.ogv uudeksi videotiedostoksi new.ogv. Uuden videon ruutukoko on asetettu 320x240 pikseliin ja bittinopeus on asetettu 512 kByte/s. Jos alkuperäisen tiedoston parametrit sopivat uusiin parametreihin, videostriimi on vain kopioitu, ilman datan muuttamista. Jos videon ruutukoko on muuttunut, se on muutettu uuteen kokoon sopivien algoritmien avulla.

Jos tiedostossa orig.ogv oli audiostriimi, se on kopioitu uuteen tiedostoon.

Audion bittinopeuden, näytenopeuden ja kanavien määrän muuttaminen:

oggResize -D64000 -F16000 -N1 orig.ogv new.ogv

Tämä komento muuttaa vain äänistriimin tiedostosta orig.ogv näytenopeuteen, joka on 16kHz, bittinopeuteen, joka on 64 kBytes/s, ja yksikanavaiseksi. Kuten videon kanssa, äänistriimi koodataan uudelleen, jos annetut parametrit eivät sovi niihin, jotka ovat vorbis-äänistriimissä. Videostriimi kopioidaan sellaisena kuin se on.

Tämä on hyvin nopea tapa koodata ainoastaan audiostreami uudelleen, ilman muita proseduureja, kuten ogg-tiedoston striimien jakamista ja uudelleen liittämistä.

Videoruudun koon muuttaminen, äänen bittinopeuden muuttaminen, kahden kuvan lisääminen videon päälle tietyksi ajanjaksoksi:

```
oggResize -s300x200 -D32000 -d1024000 -A etwas.png,2,7,s:etwasneues.png,5,10 \ orig.ogv  
new.ogv
```

Tämä komentorivi muuttaa ääni- ja videostriimin, kuten kuvattu yllä aiemmin mainituissa esimerkeissä.

Lisäksi se lisää kuvan PNG-muodossa ja alfa-kanavan **etwas.png** videoon kohdasta 2 sekuntia kohtaan 7 sekuntia, pehmeällä sisään- ja uloshäivytyksellä, ja kuva **etwasneues.png** on sijoitettu videoruudun päälle sekunnista 5 sekuntiin 10 ilman mitään feidausta.

OGG-TIEDOSTOJEN ANALYSOIMINEN

On olemassa joukko työkaluja, joilla voi analysoida Ogg-tiedostoja. Nämä ovat paketeissa **Vorbis tools**, **oggz tools** ja **Ogg Video Tools**. Aivan kuin editoinnin työkalut, nämä työkalut voivat sisältää päällekkäisiä toimintoja, tässä keskitymme kuitenkin tarjolla olevien työkalujen normaaliin käyttöön ja käyttöalueisiin.

Tässä keskustellaan seuraavista työkaluista:

Paketista **Vorbis tools**:

- ogginfo

Paketista **oggz tools**:

- oggz-info
- oggz-comment
- oggz-validate
- oggz-sort
- oggz-dump
- oggz-sort

Paketista **Ogg Video Tools**:

- oggDump
- oggLength

TIETOA OGG-TIEDOSTOSTA

Jos tahdot saada tietoa tiedoston sisällä saatavilla olevista striimeistä ja näiden striimien luonteesta, käytä **ogginfo** tai **oggz-info**.

```
$ oggz-info big_buck_bunny_480p_stereo-small.ogv
Content-Duration: 00:09:56.384
```

```
Theora: serialno 0000020573
9546 packets in 6797 pages, 1.4 packets/page, 1.037% Ogg overhead
Video-Framerate: 16.000 fps
Video-Width: 320
Video-Height: 240
```

```
Vorbis: serialno 0000003594
18641 packets in 576 pages, 32.4 packets/page, 1.412% Ogg overhead
Audio-Samplerate: 16000 Hz
Audio-Channels: 2
```

Printattu ulostulo on tässä lista Ogg-tiedostossa saatavilla olevista striimeistä ja striimien parametreista (esimerkiksi Theorassa videon kuvan koko ja kuvataajuus ja kanavanumero, Vorbisissa näytetaajuus ja kanavanumero).

ogginfo jopa printtaa vielä yksityiskohtaisempaa tietoa, kuten versiotiedot ja kaiken tiedon, jota on saatavilla videon ja audion otsikkopaketeista, kuten kuvan ja äänen otsikkopaketeista, esimerkiksi kuvan kokosuhde tai väriavaruudet.

oggz-info voi kertoa sinulle enemmän Ogg-tiedostosta. Varsinkin **-a** -optio kertoo sinulle vielä yksityiskohtaisempaa tietoa tiedostostasi. Kuitenkin tämä tieto on pääasiassa tilastollista.

Lukeaksesi tai editoidaksesi kommenttikenttiä, jotka tulevat jokaisen striimin mukana Ogg-tiedostossa. Tätä varten oggz-työkalut tarjoavat **oggz-comment** komentorivityökalun.

Esimerkkiä kommenttien printtaamisesta.

```
$ oggz-comment -l big_buck_bunny_480p_stereo-small.ogg
Theora: serialno 0000020573
Vendor: Xiph.Org libTheora I 20081020 3 2 1
ENCODER: oggvideotools-0.8-win32
Vorbis: serialno 0000003594
Vendor: Xiph.Org libVorbis I 20090514
ENCODER: oggvideotools-0.8
```

Lisätäksesi kommentin, lisää "Tag=Value" -pari komentorivin loppuun.

```
$ oggz-comment big_buck_bunny_480p_stereo-small.ogg -o bigbuck_comment.ogv \
"LICENCE=Creative Commons CC-BY-SA"
```

```
$ oggz-comment -l bigbuck_comment.ogv
Theora: serialno 0000020573
Vendor: Xiph.Org libTheora I 20081020 3 2 1
ENCODER: oggvideotools-0.8-win32
LICENCE: Creative Commons CC-BY-SA
Vorbis: serialno 0000003594
Vendor: Xiph.Org libVorbis I 20090514
ENCODER: oggVideoTools 0.8
LICENCE: Creative Commons CC-BY-SA
```

Voit lisätä kommentin vain yhteen tarjolla olevaan striimiin tai poistaa kommentteja. Ole hyvä ja katso apusivuja näistä kehittyneemmistä vaihtoehdoista.

OGG-TIEDOSTON YKSITYISKOHTAINEN ANALYYSI

Kehittäjien on usein hyödyllistä katsoa streamia itseään ja analysoida, onko Ogg-tiedosto yhteensopiva Ogg-standardin kanssa.

Tiedoston validoimiseksi voit käyttää komentorivityökalua **oggz-validate**:

```
$ oggz-validate input.ogv
input.ogv: Error:
serialno 1101839243: Terminal header page has non-zero granulepos
serialno 1101839243: Terminal header page contains non-header segment
```

Varsin yleinen ongelma on, että **oggz-validate** voi raportoida huonosti järjesteltyjä Ogg-tiedostoja. Nämä toimivat yleensä, mutta voivat aiheuttaa ongelmia, kuten ajoittaista pysähtelyä, tai lisätä muistin käyttöä. Työkalu **oggz-sort** voi auttaa korjaamaan nämä järjestysongelmat:

```
$ oggz-sort -o output.ogv input.ogv
```

Paljon tarkempi kuva saadaan työkaluilla **oggz-dump** ja **oggDump**. Molemmat näistä työkaluista kirjoittavat yksityiskohtaista ulostulotietoa konsoliin.

oggz-dump voi tuoda pakettien ulostulon kaikista tai joistain erityisistä striimeistä annetun Ogg-tiedoston sisällä.

```
$ oggz-dump myfile.ogv
[ ... ]
0b40: 6f13 abc9 f3ac 9dc2 cec5 9c62 0e70 1fab o.....b.p..
0b50: f891 01a8 0633 430e 308a 6f8c 86c3 131a .....3C.0.o.....
0b60: 3ab6 840d edab e79f fe24 a4 ... ..$.

00:00:32.104: serialno 0938763527, calc. gpos 513664, packetno 1022: 269 bytes
0000: 3eb7 e92b 50b4 0028 a93c 8bc4 0039 0948 >..+P..(<...9 H
0010: 1d58 0000 0080 5f03 db01 ee92 2482 48dd .X.....$.H.
0020: d73f 2562 feeb de81 94fe 75fb b7ff decf .?%b.....u....
0030: 18c0 b830 619f b6c5 cf0d 00f3 b5f5 bbbd ...0a.... ..
[ ... ]
```

Komennon **oggDump** avulla voit päättää, jos tahdot nähdä sivut tai pakettitiedot Ogg-tiedostosta asettamalla **-p** (paketti) tai **-g** (sivu) option. Edelleenkin voit spesifioida striimin detajitason käyttämällä **-l** (detail) asetusta. Detajit voidaan asettaa arvosta 1 (ei yksityiskohtia) arvoon 5 (eniten yksityiskohtia):

```
$ oggDump -l3 -g myVideo.ogv
```

```
[ ... ]
Ogg Page: header length = 42 and body length = 3600
Header Information:
    Ogg Version      : 0
    Serial No       : 0x37f46507
    Packet Type     : fresh packet
    Page Type      : normal page
    Last Page      : normal page
    Granule Position : 0(0x0)
    Page Number     : 1
    Checksum       : 0x5ced317c
    Table Segments  : 15

Segments:
67 ff ff ff ff ff ff ff ff ff ff ff ff ff b6

Header Hex dump:
4f 67 67 53 00 00 00 00 00 00 00 00 00 07 65
f4 37 01 00 00 00 7c 31

[ ... ]
```

VINKKEJÄ JA TEMPPUJA

Tämä osuus listaa joitain hyödyllisiä tapoja, joilla voi käyttää useita yllä kuvattuja työkaluja.

Striimin ajallinen kesto

Videon ajallisen pituuden saamiseksi selville voit käyttää komentoa **oggz-info** tai **ogginfo**, mutta tässä tapauksessa joudut hankkimaan informaation skriptillä ulostulon kautta. **oggLength** antaa sinulle toisen tavan saada pituus ulos suoraan, jotta voit käyttää sitä helpommin.

Luo Ogg Vorbis -tiedosto, joka on täynnä hiljaisuutta, ja sopii täsmälleen videoon

```
$ oggSilence -l`oggLength videoFile.ogv` -o audioSilence.oga
```

Äänileikkeen luominen videon osasta

Nämä kaksi komentoa tallentavat lyhyen viiden sekunnin äänileikkeen kohdasta, joka alkaa videon kestätyä 40 sekuntia.

```
$ oggz-rip -c vorbis video.ogv -o temp.ogg
$ oggz-chop -s 40 -e 45 -o soundbyte.ogg temp.og
```

Lisätään tekstitykset striimin sisään

Oletko tehnyt saksankielisen tekstityksen englanninkieliseen videoon, jonka löysit netistä?

```
$ kateenc -t srt -l de -c SUB -o subtitles.ogg subtitles.srt
$ oggz-merge -o video-with-german-subtitles.ogv original-video.ogv subtitles.ogg
```

Kaikkien striimin pakettien saaminen hyvin pakattuna

```
$ oggz-dump video.ogv | grep packetno | less

00:00:00.437: serialno 0000020573, calc. gpos 0|7, packetno 10: 414 bytes
00:00:00.500: serialno 0000020573, calc. gpos 0|8, packetno 11: 692 bytes
00:00:00.562: serialno 0000020573, calc. gpos 0|9, packetno 12: 600 bytes
00:00:00.625: serialno 0000020573, granulepos 0|10, packetno 13: 859 bytes
00:00:00.687: serialno 0000020573, calc. gpos 0|11, packetno 14: 700 bytes
00:00:00.750: serialno 0000020573, calc. gpos 0|12, packetno 15: 1.154 kB
00:00:00.812: serialno 0000020573, calc. gpos 0|13, packetno 16: 878 bytes
00:00:00.875: serialno 0000020573, granulepos 0|14, packetno 17: 1.342 kB
00:00:00.000: serialno 0000003594, calc. gpos 0, packetno 3: 118 bytes
00:00:00.032: serialno 0000003594, calc. gpos 512, packetno 4: 115 bytes
00:00:00.064: serialno 0000003594, calc. gpos 1024, packetno 5: 127 bytes
```

TEKIJÄT

44. Lisenssi

44. LISENSSI

Kaikki kappaleet ovat kirjoittajien tekijänoikeuden alaisia. Jos muuten ei sanota, kaikki luvut tässä käyttöoppaassa on lisensoitu **GNU General Public License version 2** mukaisesti.

Tämä dokumentaatio on vapaata dokumentaatiota: voit jakaa sitä eteenpäin ja/tai muokata sitä Free Software Foundationin GNU General Public License mukaisesti; joko lisenssin version 2, tai (tahtoeassasi) minkä tahansa myöhemmän version.

Dokumentaatiota jaellaan siinä toivossa, että se on käyttökelpoisa, mutta **ILMAN MITÄÄN TAKUUTA**; ilman edes **MYYTÄVYYDEN** tai **TIETTYYN KÄYTTÖÖN SOPIVUUDEN** oletettua takuuta. Katso lisätietoja GNU General Public Licensestä.

Tämän dokumentaation mukana olisi pitänyt tulla kopio GNU General Public Licensestä, mikäli sitä ei tullut kirjoita osoitteeseen Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

SUOMENKIELISEN VERSION KÄÄNTÄJÄT JA KIRJOITTAJAT

TÄSTÄ KÄYTTÖOPPAASTA

© Tomi Toivio 2009, 2010

OGGIN ANALYSOIMINEN

© Tomi Toivio 2009, 2010

VIRTAUSTOISTON TALLENTAMINEN

© Tomi Toivio 2009, 2010

Modifications:

Tomi Toivio 2009

TIEDOSTOJEN YHDISTÄMINEN

© Tomi Toivio 2009, 2010

KOODEKIT

© Tomi Toivio 2009, 2010

Modifications:

TWikiGuest 2009

SÄILÖT

© Tomi Toivio 2009, 2010, 2011

JAKELU

© Tomi Toivio 2009, 2010

TEE ESIKATSELUVIDEO

© Tomi Toivio 2009, 2010

TEE DIAESITYS

© Tomi Toivio 2009, 2010

TEE ESIKATSELUKUVIA

© Tomi Toivio 2009, 2010

TEKIJÄT

© adam hyde 2006, 2007

Modifications:
Tomi Toivio 2009, 2010

TIEDOSTON LEIKKAAMINEN
© Tomi Toivio 2009, 2010

JOHDANTO
© Tomi Toivio 2009, 2010

TEKSTITYKSIEN LIITTÄMINEN
© Tomi Toivio 2009, 2010

JOHDANTO
© Tomi Toivio 2009, 2010
Modifications:
Tomi Toivio 2009

FFMPEG2Theora
© Tomi Toivio 2009, 2010
Modifications:
Tomi Toivio 2009

FFMPEG2THEORA
© Tomi Toivio 2009, 2010
Modifications:
Tomi Toivio 2009

FIREFOGG
© Tomi Toivio 2009, 2010
Modifications:
Tomi Toivio 2009

HTML5
© Tomi Toivio 2009, 2010
Modifications:
Tomi Toivio 2009

HOTELLEJA THEORALLE
© Tomi Toivio 2009, 2010
Modifications:
Tomi Toivio 2009

ICECAST
© Tomi Toivio 2009, 2010

JOHDANTO
© adam hyde 2006, 2007
Modifications:
Tomi Toivio 2009, 2010
TWikiGuest 2009
Tomi Toivio 2009

TIEDOSTOJEN LIITTÄMINEN
© Tomi Toivio 2009, 2010

JOHDANTO
© Tomi Toivio 2009, 2010

MIRO

© Tomi Toivio 2009, 2010

Modifications:

Tomi Toivio 2009

THEORA SIVUILLASI

© Tomi Toivio 2009, 2010

Modifications:

Tomi Toivio 2009

KOON MUUTTAMINEN

© Tomi Toivio 2009, 2010

JOHDANTO

© Tomi Toivio 2009, 2010

Modifications:

Tomi Toivio 2009

PITIVI

© Tomi Toivio 2009, 2010

KUINKA THEORAA TOISTETAAN

© Tomi Toivio 2009, 2010

TEKSTITYKSIEN TOISTAMINEN

© Tomi Toivio 2009, 2010

JULKAISEMINEN

© Tomi Toivio 2009, 2010

Modifications:

Tomi Toivio 2009

TIEDOSTON JAKAMINEN

© Tomi Toivio 2009, 2010

JOHDANTO

© Tomi Toivio 2009, 2010

Modifications:

Tomi Toivio 2009

TSS

© Tomi Toivio 2009, 2010

Modifications:

Tomi Toivio 2009

THEORAN TEORIA

© Tomi Toivio 2009, 2010

THOGGEN

© Tomi Toivio 2009, 2010

Modifications:

Tomi Toivio 2009

VLC

© Tomi Toivio 2009, 2010

Modifications:

Tomi Toivio 2009

VLC

© Tomi Toivio 2009, 2010

Modifications:
Tomi Toivio 2009

VLC
© Tomi Toivio 2009, 2010

MITÄ ON SUORATOISTO
© Tomi Toivio 2009, 2010
Modifications:
Tomi Toivio 2009

MITÄ THEORA ON?
© Tomi Toivio 2009, 2010
Modifications:
Tomi Toivio 2009

MITÄ VIDEO ON?
© Tomi Toivio 2009, 2010

JOHDANTO
© Tomi Toivio 2009, 2010

ENGLANNINKIELISEN VERSION TEKIJÄT

ABOUT THIS MANUAL
© adam hyde 2009
Modifications:
David Kühling 2009
William Merriam 2009

ACCESSIBILITY
© TWikiGuest 2009
Modifications:
adam hyde 2009
Silvia Pfeiffer 2009

ANALYSING OGG
© Mark Hancock 2009
Modifications:
adam hyde 2009
Jan Gerber 2009
ogg k 2009
Jörn Seger 2009

STREAM RIPPING
© adam hyde 2007, 2009
Modifications:
Alan Toner 2009
Edward Cherlin 2009
Holmes Wilson 2009
Jan Gerber 2009
ogg k 2009
Thomas Middleton 2008
Jörn Seger 2009

CAT FILES

© adam hyde 2009

Modifications:

Robert Valliant 2009

Jörn Seger 2009

CODECS

© TWikiGuest 2008

Modifications:

adam hyde 2009

Holmes Wilson 2009

Jaap Hoetmer 2009

Jan Gerber 2009

ogg k 2009

Robert Valliant 2009

susanne lang 2009

Jörn Seger 2009

CONTAINERS

© TWikiGuest 2008

Modifications:

adam hyde 2009

Holmes Wilson 2009

Jan Gerber 2009

ogg k 2009

susanne lang 2009

Jörn Seger 2009

DISTRIBUTION

© adam hyde 2007, 2009

Modifications:

Holmes Wilson 2009

ogg k 2009

susanne lang 2009

Thomas Middleton 2008

CREATE PREVIEW

© Jörn Seger 2009

Modifications:

adam hyde 2009

Alan Toner 2009

David Kühling 2009

CREATE SLIDESHOWS

© Jörn Seger 2009

Modifications:

adam hyde 2009

Alan Toner 2009

David Kühling 2009

Edward Cherlin 2009

CREATE THUMBNAILS

© Jörn Seger 2009

Modifications:

adam hyde 2009

Alan Toner 2009

David Kühling 2009

Mark Hancock 2009

CREDITS

© adam hyde 2006, 2007, 2009

CUT A FILE

© adam hyde 2009

Modifications:

David Kühling 2009

Edward Cherlin 2009

Robert Valliant 2009

INTRODUCTION

© adam hyde 2007, 2009

Modifications:

Alan Toner 2009

Edward Cherlin 2009

Holmes Wilson 2009

Jan Gerber 2009

salsa man 2009

susanne lang 2009

Thomas Middleton 2008

Jörn Seger 2009

EMBEDDING SUBTITLES

© adam hyde 2009

Modifications:

Edward Cherlin 2009

Holmes Wilson 2009

Jan Gerber 2009

susanne lang 2009

INTRODUCTION

© TWikiGuest 2008

Modifications:

adam hyde 2009

David Kühling 2009

Jan Gerber 2009

jay maechtlen 2009

Jörn Seger 2009

FFMPEG2THEORA

© TWikiGuest 2008

Modifications:

adam hyde 2009

David Kühling 2009

Edward Cherlin 2009

Holmes Wilson 2009

Jan Gerber 2009

ogg k 2009

Jörn Seger 2009

FFMPEG2THEORA

© adam hyde 2007, 2009

Modifications:

Alan Toner 2009

Holmes Wilson 2009

Jan Gerber 2009

Thomas Middleton 2008

Jörn Seger 2009

FIREFOGG

© maxigas maxigas 2007

Modifications:

adam hyde 2009

David Kühling 2009

Jan Gerber 2009

Peter W 2009

Thomas Middleton 2008

Jörn Seger 2009

GLOSSARY

© adam hyde 2006, 2009

Modifications:

Edward Cherlin 2009

Holmes Wilson 2009

Jan Gerber 2009

ogg k 2009

Thomas Middleton 2008

HTML5

© Jörn Seger 2009

Modifications:

adam hyde 2009

Ali Gunduz 2009

Andrew Nicholson 2009

Jan Gerber 2009

Michael Dale 2009

ogg k 2009

Robert Valliant 2009

Silvia Pfeiffer 2009

HOSTING SITES

© TWikiGuest 2008

Modifications:

adam hyde 2009

Andrew Nicholson 2009

Holmes Wilson 2009

Jan Gerber 2009

ICECAST

© adam hyde 2006, 2007, 2008

Modifications:

Zita Joyce 2008

INTERESTING EXTRAS

© TWikiGuest 2009

Modifications:

adam hyde 2009

Jan Gerber 2009

Silvia Pfeiffer 2009

INTRODUCTION

© adam hyde 2006, 2007, 2009

Modifications:

Brylie Oxley 2009

David Kühling 2009

Edward Cherlin 2009
Holmes Wilson 2009
Jan Gerber 2009
Maik Merten 2009
ogg k 2009
susanne lang 2009
Thomas Middleton 2008

JOIN FILES

© adam hyde 2009
Modifications:
David Kühling 2009
Robert Valliant 2009

INTRODUCTION

© TWikiGuest 2008
Modifications:
adam hyde 2009
David Kühling 2009
Edward Cherlin 2009
Holmes Wilson 2009
ogg k 2009
Robert Valliant 2009
Jörn Seger 2009

MIRO

© Jörn Seger 2008
Modifications:
adam hyde 2009
Holmes Wilson 2009
TWikiGuest 2008
Thom Hastings 2009

HOSTING THEORA YOURSELF

© TWikiGuest 2008
Modifications:
adam hyde 2009
Chris Double 2009
Holmes Wilson 2009
Jan Gerber 2009
susanne lang 2009

RESIZING

© Jan Gerber 2009
Modifications:
adam hyde 2009
Robert Valliant 2009
Jörn Seger 2009

INTRODUCTION

© TWikiGuest 2008
Modifications:
adam hyde 2009
Jan Gerber 2009
ogg k 2009
susanne lang 2009
Jörn Seger 2009

PITIVI

© adam hyde 2006, 2009

Modifications:

Edward Cherlin 2009

Holmes Wilson 2009

Jan Gerber 2009

Thomas Middleton 2008

HOW TO PLAY THEORA

© TWikiGuest 2008

Modifications:

adam hyde 2009

Gergely Máté 2009

Holmes Wilson 2009

Jan Gerber 2009

ogg k 2009

Robert Valliant 2009

Stjepan Rajko 2009

susanne lang 2009

PLAYING SUBTITLES

© adam hyde 2009

Modifications:

Edward Cherlin 2009

Holmes Wilson 2009

susanne lang 2009

PUBLISHING

© adam hyde 2007, 2009

Modifications:

another sam 2009

Edward Cherlin 2009

Holmes Wilson 2009

Jan Gerber 2009

Mark Grandi 2009

susanne lang 2009

Thomas Middleton 2008

SPLIT A FILE

© adam hyde 2009

Modifications:

David Kühling 2009

ogg k 2009

Robert Valliant 2009

Jörn Seger 2009

INTRODUCTION

© ogg k 2009

Modifications:

adam hyde 2009

Edward Cherlin 2009

Holmes Wilson 2009

susanne lang 2009

TSS

© TWikiGuest 2008

Modifications:

adam hyde 2009

Alan Toner 2009
alejo duque 2009
Holmes Wilson 2009

THEORA THEORY

© Jörn Seger 2008, 2009
Modifications:
adam hyde 2008, 2009
Alan Toner 2009
David Kühling 2009
Robert Valliant 2009

THOGGEN

© adam hyde 2007, 2009
Modifications:
David Kühling 2009
Edward Cherlin 2009
Jan Gerber 2009
ogg k 2009
Jörn Seger 2009

VLC

© TWikiGuest 2008
Modifications:
adam hyde 2009
David Kühling 2009
Dennis J 2009
Mark Hancock 2009
Jörn Seger 2009

VLC

© TWikiGuest 2008
Modifications:
adam hyde 2009
Holmes Wilson 2009
Jan Gerber 2009

VLC

© adam hyde 2007, 2009
Modifications:
Alan Toner 2009
Edward Cherlin 2009
jay maechtlen 2009
Thomas Middleton 2008
Jörn Seger 2009

WHAT IS STREAMING?

© TWikiGuest 2008
Modifications:
adam hyde 2009
Alan Toner 2009
Jörn Seger 2009

WHAT IS THEORA?

© TWikiGuest 2008
Modifications:
adam hyde 2009
Edward Cherlin 2009

Holmes Wilson 2009
jay maechtlen 2009
Joshua Gay 2009
Lachlan Musicman 2009
susanne lang 2009

WHAT IS VIDEO?

© TWikiGuest 2008
Modifications:
adam hyde 2009
Brylie Oxley 2009
Edward Cherlin 2009
Holmes Wilson 2009
Lachlan Musicman 2009
susanne lang 2009

INTRODUCTION

© Mark Hancock 2009
Modifications:
adam hyde 2009
Alan Toner 2009
David Kühling 2009
susanne lang 2009
Jörn Seger 2009



Vapaat käyttöoppaat vapaille ohjelmille

GNU GENERAL PUBLIC LICENSE Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that

you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in

themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues),

conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this license incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

